

**SWORD**

XD-LAB-ARCH-007

# Lab7: Pipelined CPU Accessing Mem in Multiple Cycle

Joseph Xu


2019-1-23

## 修改记录

版本号.	作者	描述	修改日期
1.0	Joseph Xu	初稿	2019-01-23


## 审核记录

姓名	职务	签字	日期

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	1 of 35
	作者	修改日期	公开	
	Joseph Xu	2019/1/23		

## 目录

修改记录 .....	0
审核记录 .....	1
1. 实验准备 .....	5
1.1 实验目的 .....	5
1.2 实验内容 .....	5
1.3 实验资料 .....	5
2. 实验步骤 .....	7
2.1 新建工程 .....	7
2.1 修改文件 .....	9
2.2 功能仿真 .....	21
2.3 生成 bitstream 文件 .....	30
2.4 下载验证 .....	31

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	2 of 35
	作者	修改日期	公开	
	Joseph Xu	2019/1/23		

## 图目录

图 1-1	实验资料网站 .....	5
图 1-2	实验文档和工程源文件 .....	6
图 2-1	运行 Vivado .....	7
图 2-2	打开工程 .....	7
图 2-3	打开 Lab06 工程 .....	8
图 2-4	工程另存为 .....	8
图 2-5	工程另存为 lab07 .....	9
图 2-6	需要修改的 lab07 工程 .....	9
图 2-7	点击 Expand All 快捷按钮 .....	10
图 2-8	mips_top 展开后的文件层次 .....	10
图 2-9	移除所有旧文件后的源码层次 .....	11
图 2-10	添加新文件 .....	11
图 2-11	增加 design sources .....	12
图 2-12	add files .....	12
图 2-13	切换目录 .....	13
图 2-14	选择文件 .....	13
图 2-15	lab07 添加的文件 .....	14
图 2-16	继续添加 hex 文件 .....	14
图 2-17	Next .....	15
图 2-18	添加文件 .....	15
图 2-19	指定全部类型文件可见 .....	16
图 2-20	切换目录 .....	16
图 2-21	选择 hex 文件 .....	17
图 2-22	增加到工程 .....	17
图 2-23	设置文件类型 .....	18
图 2-24	设置类型为 Data Files .....	18
图 2-25	设置文件类型 .....	19
图 2-26	设置类型为 Data Files .....	19
图 2-27	Refresh Hierarchy .....	20
图 2-28	Data Files .....	20
图 2-29	添加所有文件后的文件列表 .....	21



	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	3 of 35
	作者	修改日期	公开	
	Joseph Xu	2019/1/23		

图 2-30	设置仿真顶层.....	22
图 2-31	开始仿真.....	22
图 2-32	仿真视图.....	23
图 2-33	仿真所需要观察的所有信号.....	24
图 2-34	菜单中选择“Open Waveform Configuration” .....	25
图 2-35	切换目录.....	25
图 2-36	选择 wcfg 文件.....	25
图 2-37	通过加载波形配置文件添加信号 .....	26
图 2-38	Restart Simulation.....	26
图 2-39	Run for... .....	27
图 2-40	仿真结果 1.....	28
图 2-41	仿真结果 2.....	29
图 2-42	关闭仿真视图.....	29
图 2-43	生成 bitstream.....	30
图 2-44	自动运行中间步骤 .....	30
图 2-45	生成 bitstream 成功 .....	31
图 2-46	连接 SWORD4.0.....	31
图 2-47	主要控制 IO .....	32
图 2-48	进入 Hardware Manager .....	32
图 2-49	自动连接到 SWORD4.0 .....	33
图 2-50	Program Device.....	33
图 2-51	下载 bit 文件 .....	34
图 2-52	运行效果.....	34

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle 作者 Joseph Xu	XD-LAB-ARCH-007 修改日期 2019/1/23	1.0	4 of 35 公开

## 1. 实验准备

### 1.1 实验目的

理解 CPU 在多时钟周期内访问 Memory 的原理机制及该机制对 CPU 的影响。掌握为流水线 CPU 实现多时钟周期访问 memory 的设计及验证方法。并在实验 6 的基础上，修改 Memory 访问机制为多个时钟周期访问时间，CPU 增加 CPU Tick 计算功能，并测试实际循环指令执行效果，分析实际存储器对 CPU 性能的影响。

### 1.2 实验内容

本实验内容包括：

- 重新设计 Inst. ROM 和 Data RAM；
- 修改设计 CPU 的数据通路 Datapath 和控制单元 Controller；
- 对设计的 CPU 进行行为仿真，检查功能正确性；
- 上板验证并观察 CPU 的执行过程。

### 1.3 实验资料

实验资料可以从网站 <https://gitee.com/SWORDfpga/ComputerArchitecture> 获得。

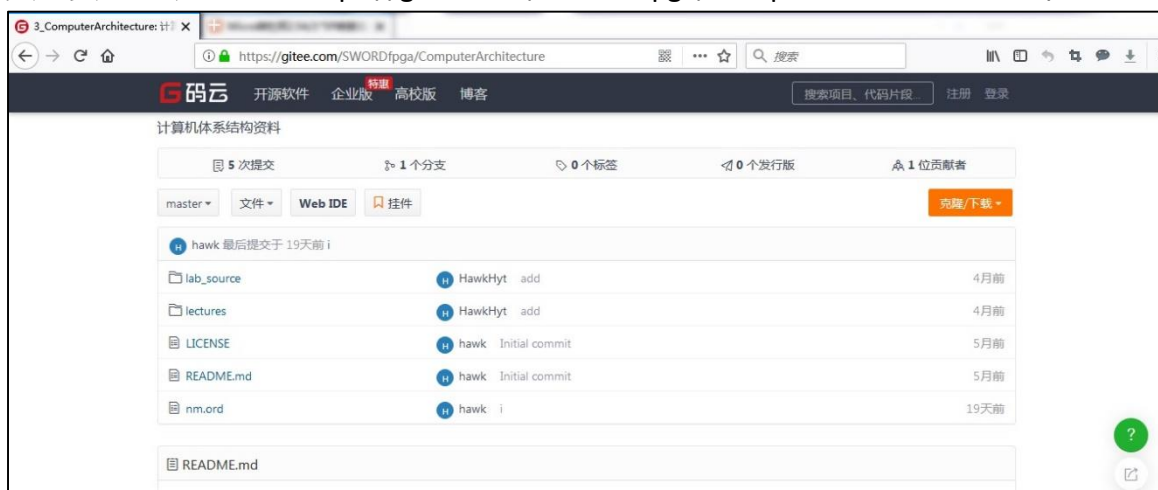



图 1-1 实验资料网站

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	5 of 35
	作者	修改日期	公开	
	Joseph Xu	2019/1/23		

为方便使用，建议建立 c:\archlabs 目录，放置实验资料和实验工程。

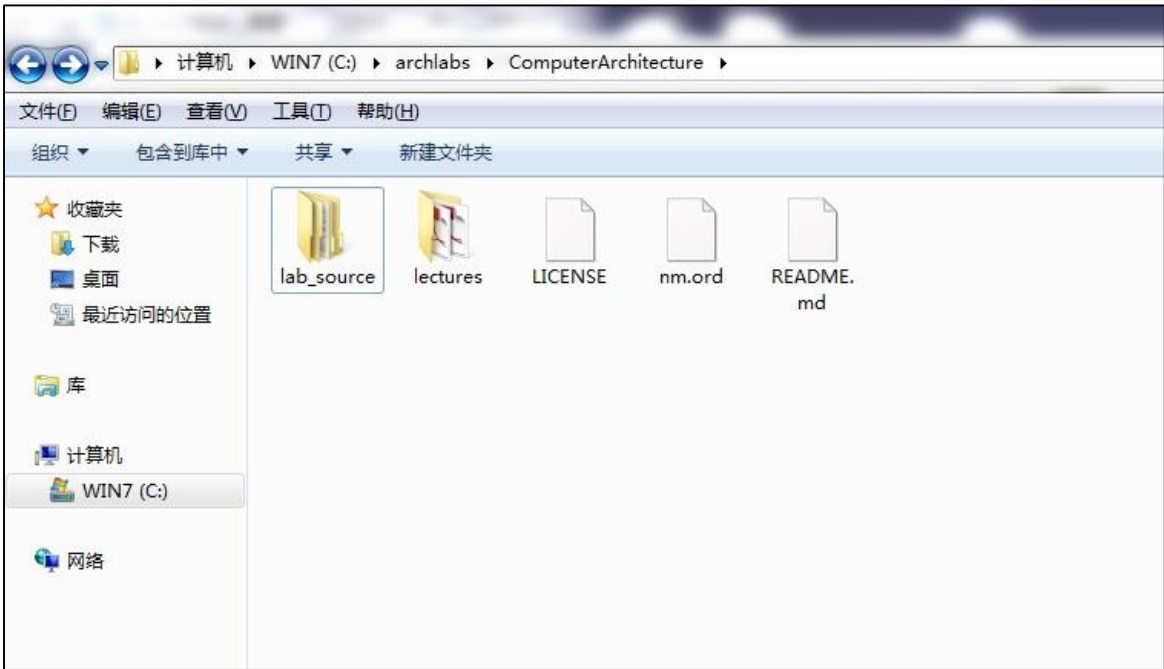



图 1-2 实验文档和工程源文件

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	6 of 35
	作者	修改日期	公开	
	Joseph Xu	2019/1/23		

2. 实验步骤

2.1 新建工程

首先启动 Vivado 2014.4 开发工具，如图 2-1 所示：

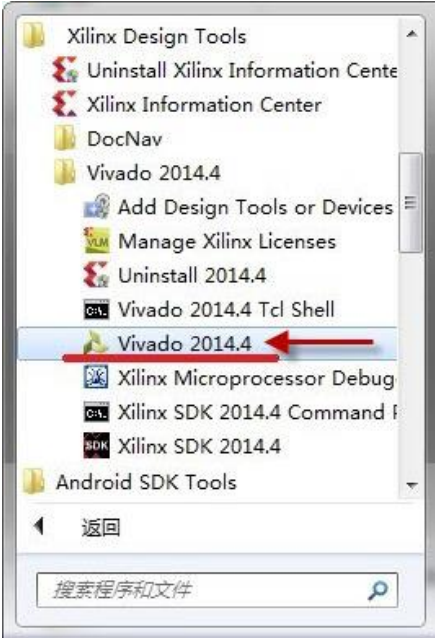


图 2-1 运行 Vivado

点击 Open Project，如图 2-2 所示：

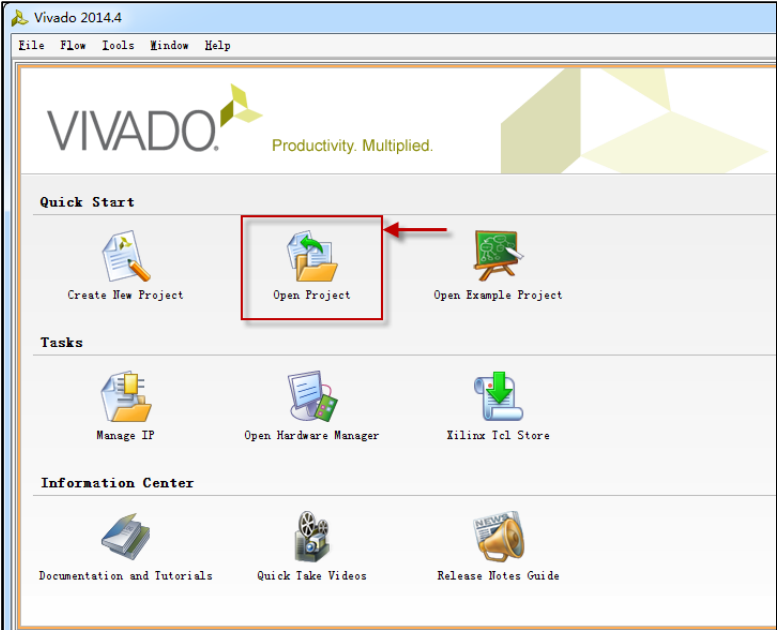



图 2-2 打开工程

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	7 of 35
	作者	修改日期	公开	
	Joseph Xu	2019/1/23		



选择之前已经完成的 Lab06 工程，点击 OK 打开工程，如图 2-3 所示：

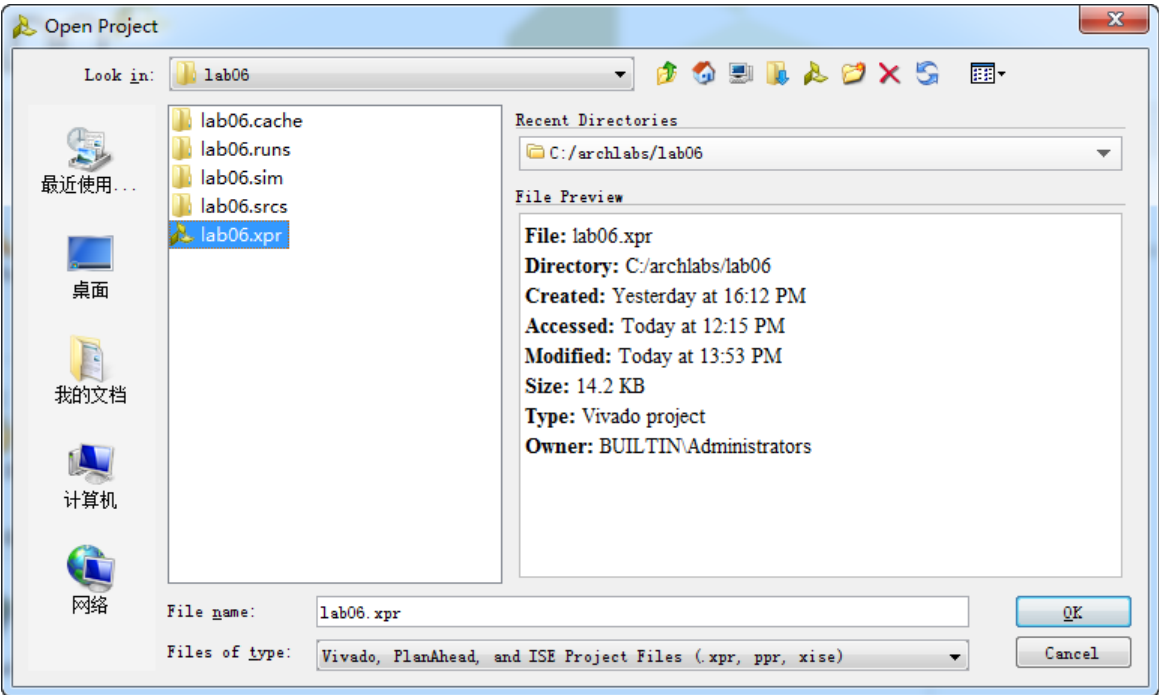


图 2-3 打开 Lab06 工程

本实验以 Lab06 工程为基础，在 File 菜单点击 **Save Project As** 将工程另存为，如图 2-4 所示：

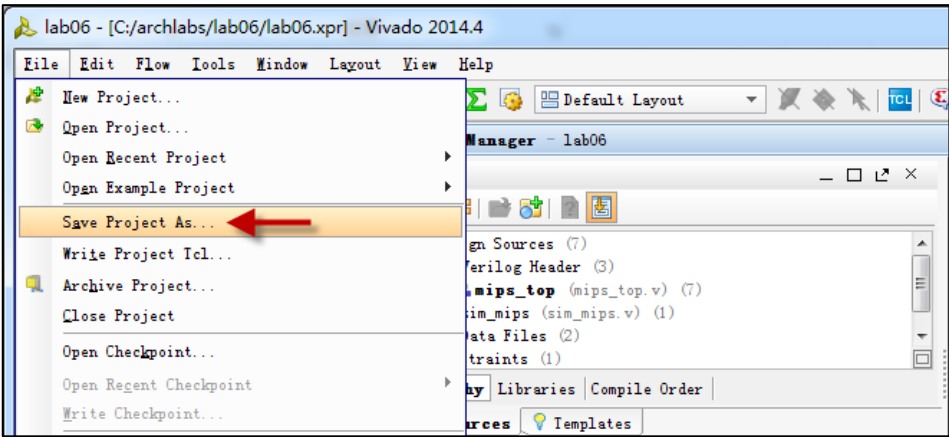



图 2-4 工程另存为

如图 2-5 所示，输入新工程名 **lab07**，选中 **Create project subdirectory**，点击 **OK** 继续。

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	8 of 35
	作者	修改日期	公开	
	Joseph Xu	2019/1/23		

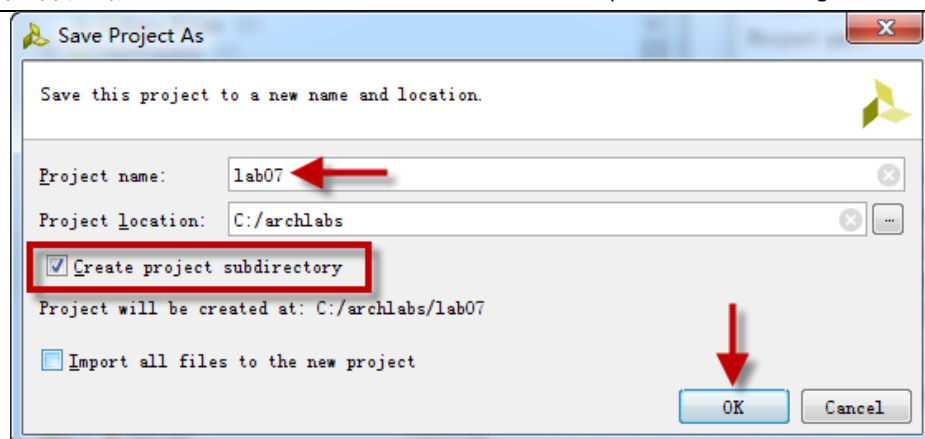


图 2-5 工程另存为 lab07

如图 2-6 所示，工程被另存为 lab07，本实验将以此工程为基础进行修改。

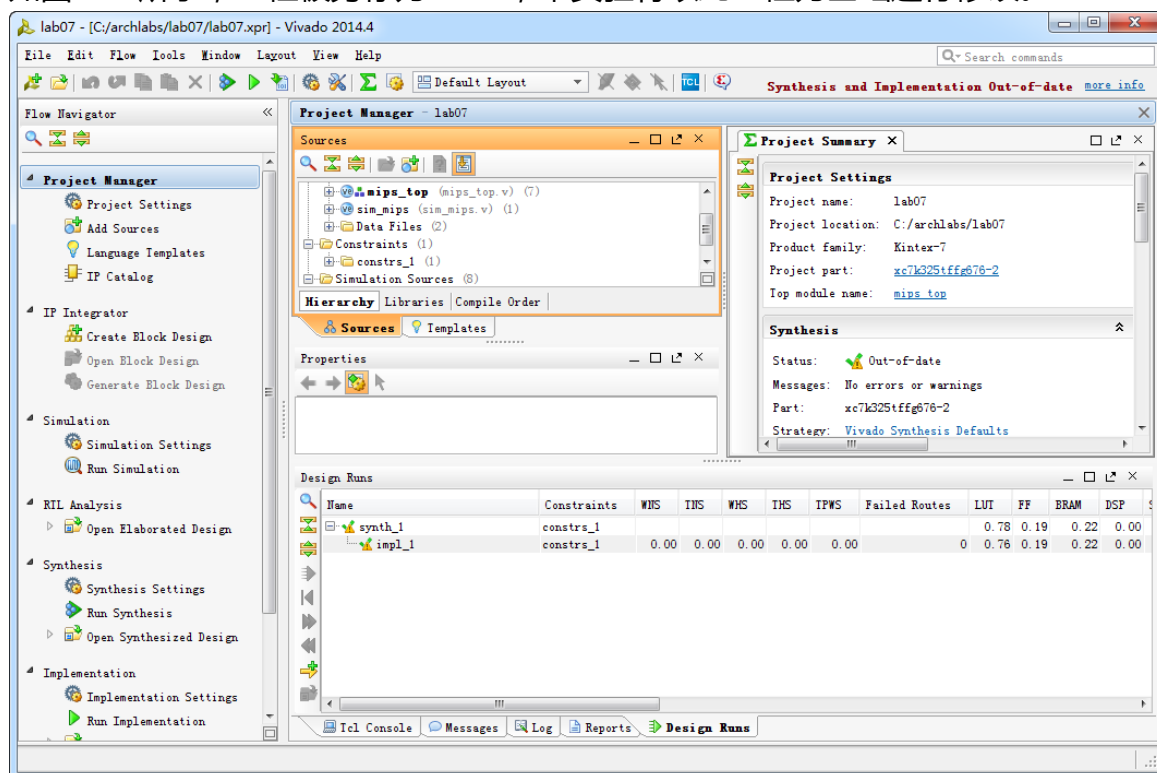



图 2-6 需要修改的 lab07 工程

## 2.1 修改文件

在 Project Manager 窗口下，首先点击 mips\_top，然后点击如图 2-7 所示的 Expand All 快捷按钮展开 mips\_top 下的所有文件：

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	9 of 35
	作者	修改日期	公开	
	Joseph Xu	2019/1/23		

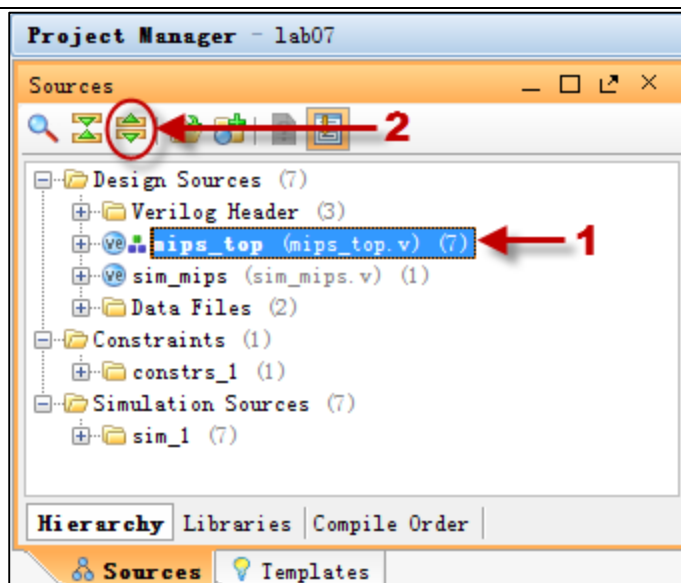


图 2-7 点击 Expand All 快捷按钮

展开后的层次如图 2-8 所示：

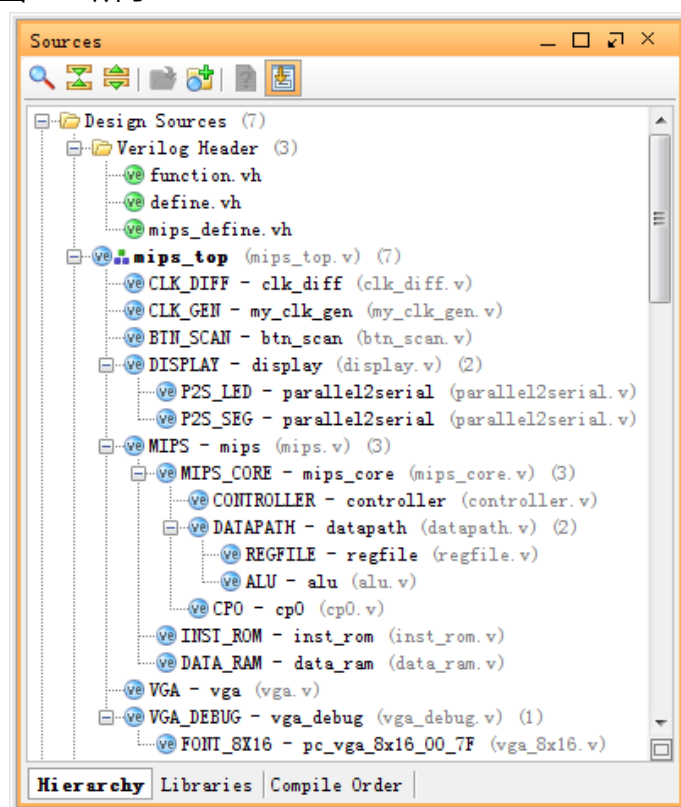



图 2-8 mips\_top 展开后的文件层次

由于本实验在 lab06 的基础上需要修改和新增一些文件，所以我们进行一些移除旧文件，添加更新文件的操作。

首先按照 lab06 里面移除文件的方法，将下列文件依次移除：

- 1) inst\_mem.hex

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	10 of 35
	作者	修改日期	公开	
	Joseph Xu	2019/1/23		

- 2) data\_mem.hex
- 3) cp0.v
- 4) inst\_rom.v
- 5) data\_ram.v
- 6) datapath.v
- 7) controller.v
- 8) mips\_core.v
- 9) mips.v
- 10) mips\_define.vh
- 11) sim\_mips.v

移除后的文件列表应如图 2-9 所示，**请仔细比对该图，确保所有旧文件都已经移除！！**

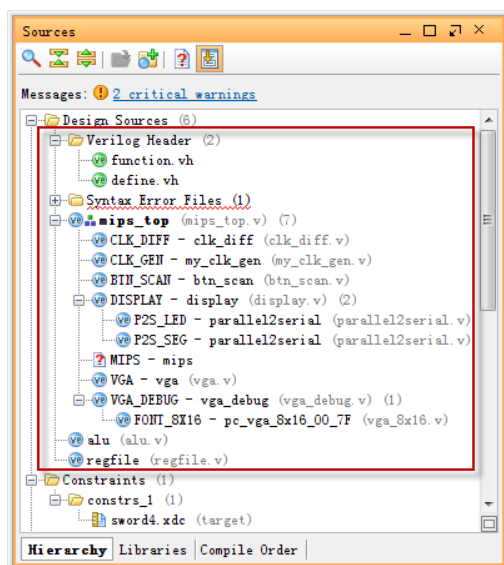


图 2-9 移除所有旧文件后的源码层次

接着在 Sources 窗口内点击 **Add Sources** 快捷按钮，添加新文件，如图 2-10 所示。

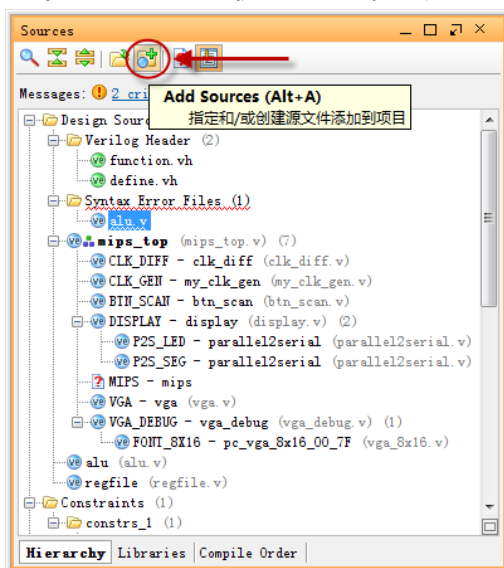



图 2-10 添加新文件

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	11 of 35
	作者	修改日期	公开	
	Joseph Xu	2019/1/23		

选中 **Add or create design sources** , 点击 **Next** , 如图 2-11 所示 :

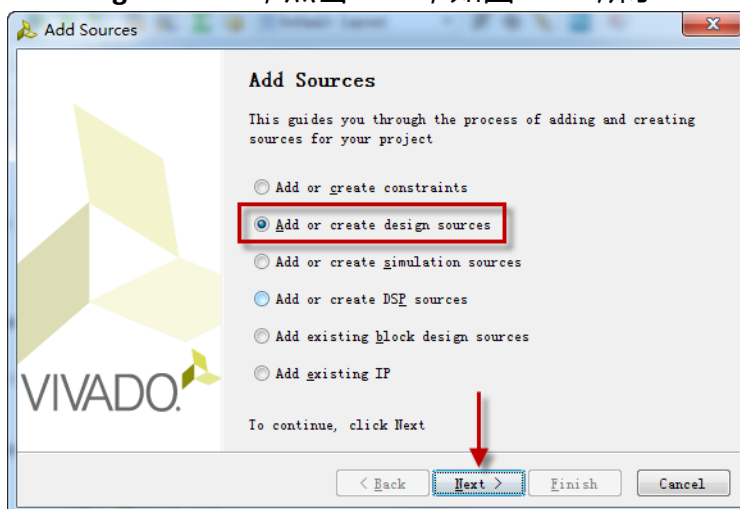


图 2-11 增加 design sources

点击 **Add Files** , 如图 2-12 所示 :

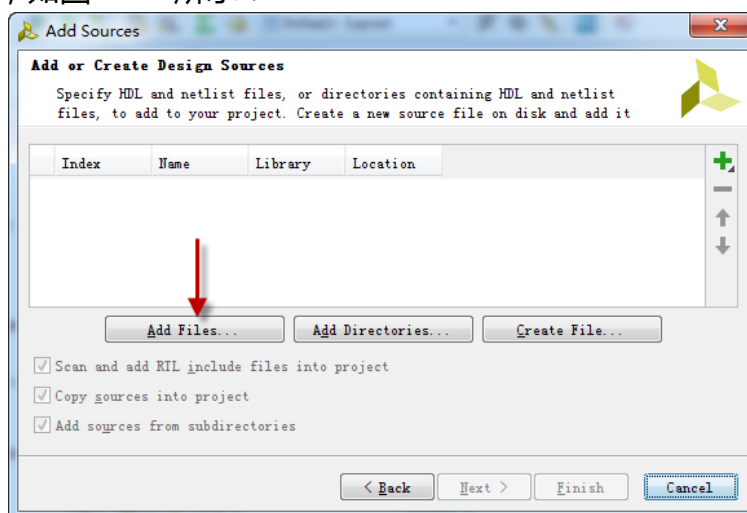



图 2-12 add files

进入 C:\archlabs\ComputerArchitecture\lab\_source\lab07\sources\cpu 目录 , 如图 2-13 所示 :

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	12 of 35
	作者	修改日期	公开	
	Joseph Xu	2019/1/23		

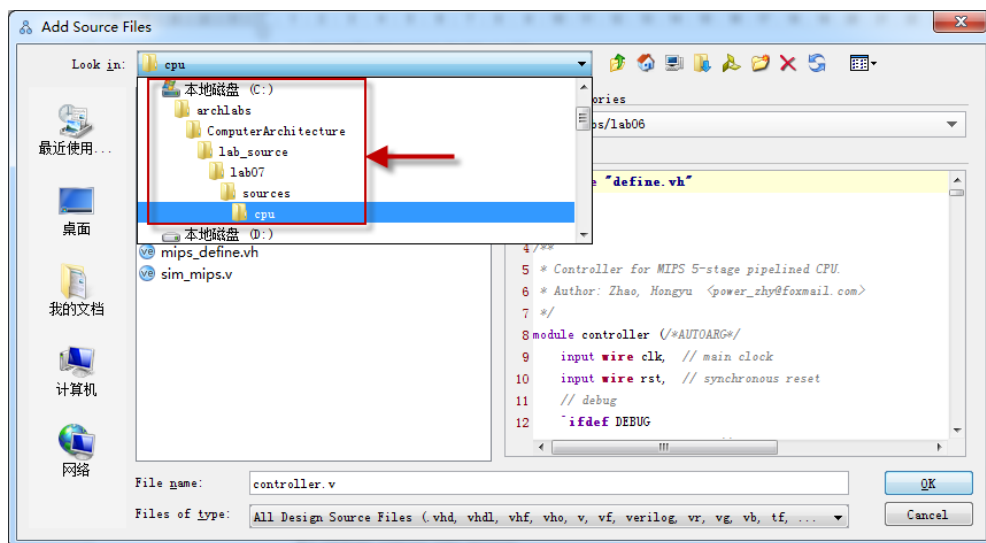


图 2-13 切换目录

选中所有文件，点击 OK 确认添加，如图 2-14 所示。

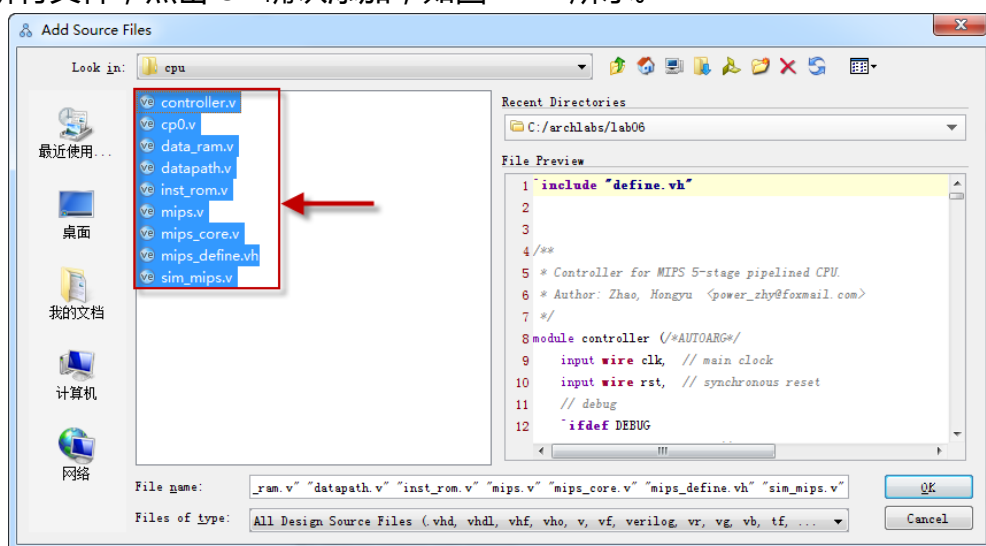


图 2-14 选择文件


如图 2-15 所示，列表中会显示要添加到工程中的文件，本次操作添加了 9 个文件，请确认添加文件的数量正确。选中 **Scan and add RTL include files into project** 和 **Copy sources into project**，点击 **Finish**。

本次操作新加入了一下文件：

协处理器文件 cp0.v 中增加了 CPU Tick 计算功能；

增加的 alu.v、controller.v 和 datapath.v 修改了对 CP0 的响应控制和数据通路；

mips.v 和 mips\_core.v 作为连接 controller 和 datapath 的模块也做了对应修改；

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	13 of 35
	作者	修改日期	公开	
	Joseph Xu	2019/1/23		

mips\_define.vh 里增加了对 CPU Tick 指令的支持定义；  
sim\_mips.v 里对应的仿真顶层文件。

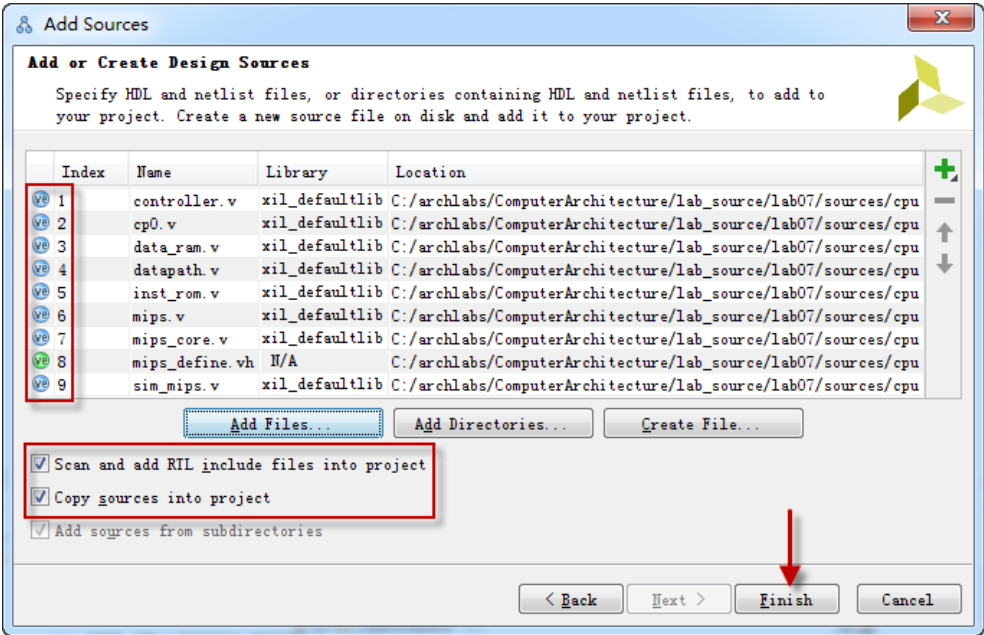


图 2-15 lab07 添加的文件

接着我们添加 hex 文件，同样点击如图 2-16 所示的 Add Sources 快捷按钮：

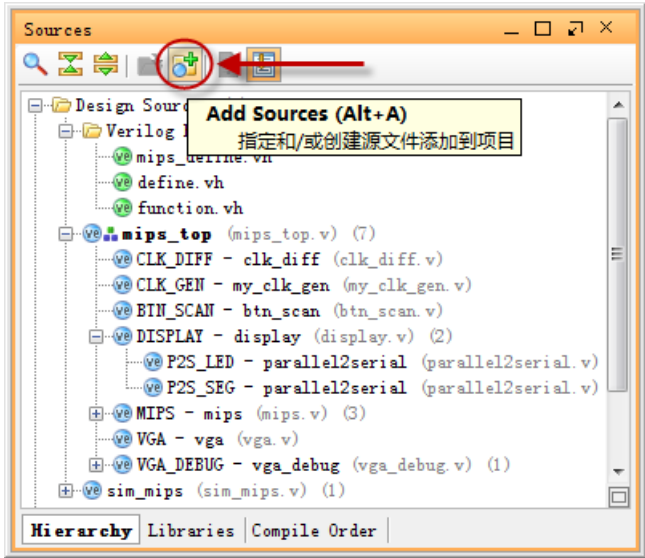


图 2-16 继续添加 hex 文件

选中 Add or create design sources，点击 Next，如图 2-17 所示：


	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	14 of 35
	作者	修改日期	公开	
	Joseph Xu	2019/1/23		



图 2-17 Next

用 Add Files 添加文件，如图 2-18 所示：

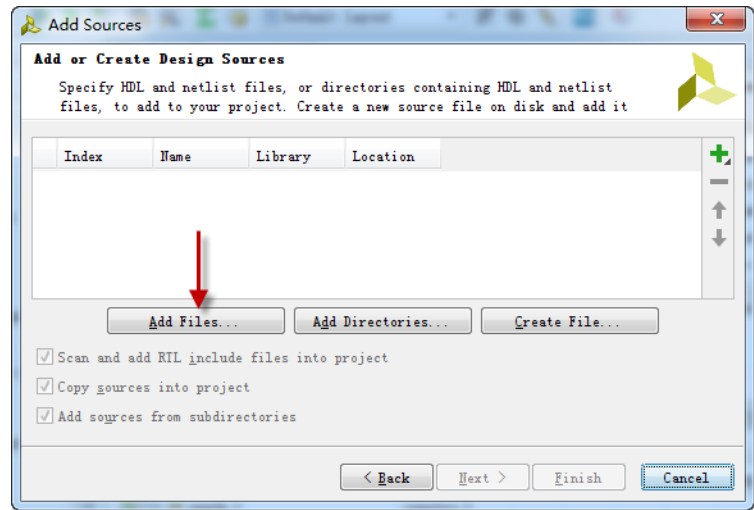



图 2-18 添加文件

指定文件类型为 All Files，如图 2-19 所示：

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	15 of 35
	作者	修改日期	公开	
	Joseph Xu	2019/1/23		



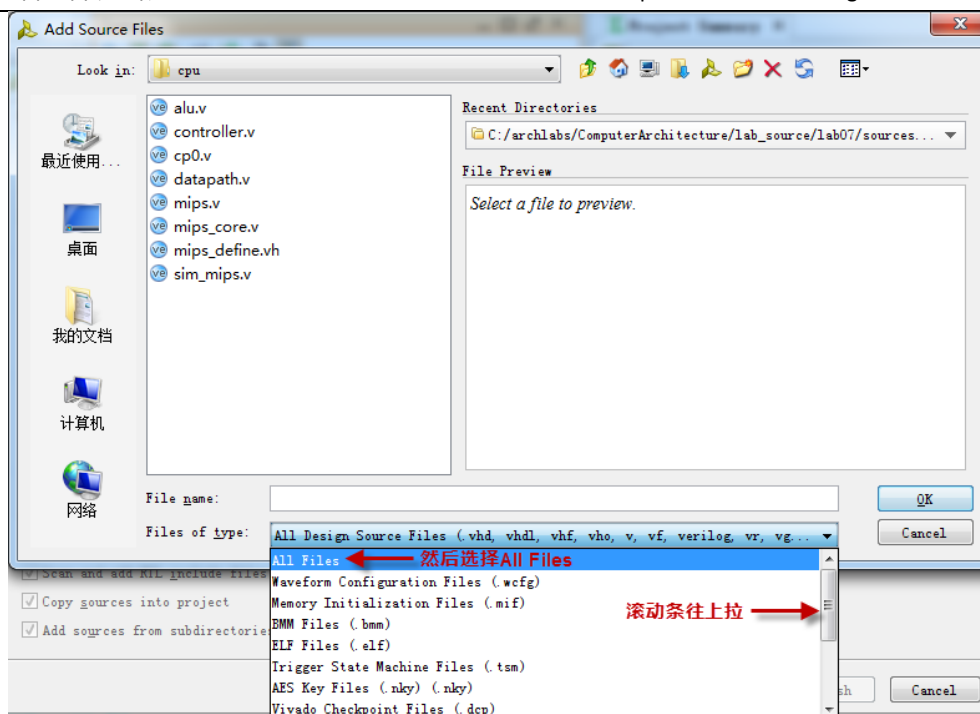


图 2-19 指定全部类型文件可见

切换到 C:\archlabs\ComputerArchitecture\lab\_source\lab07\data 目录，如图 2-20 所示：

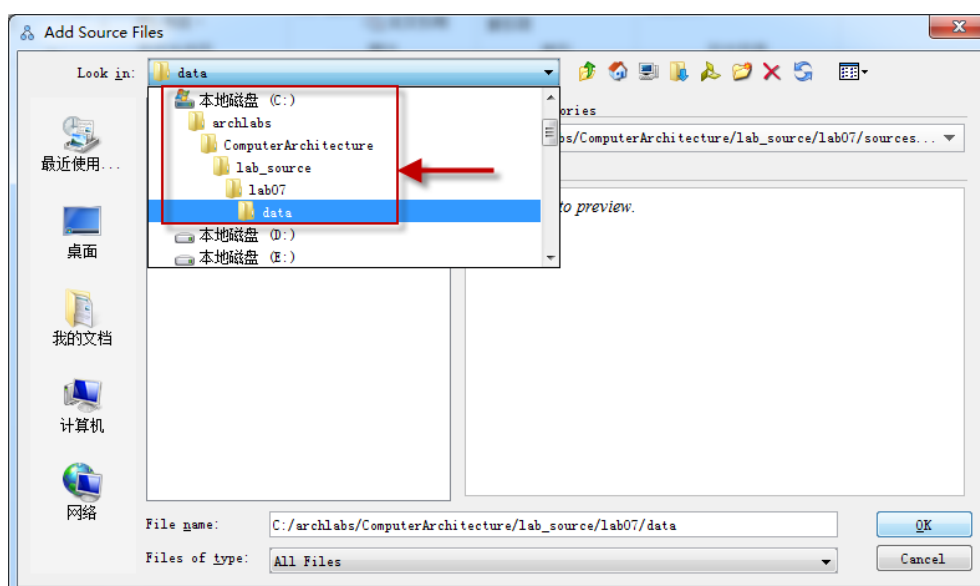



图 2-20 切换目录

选择 data\_mem.hex 和 inst\_mem.hex 文件，如图 2-21 所示：

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	16 of 35
	作者	修改日期	公开	
Joseph Xu		2019/1/23		

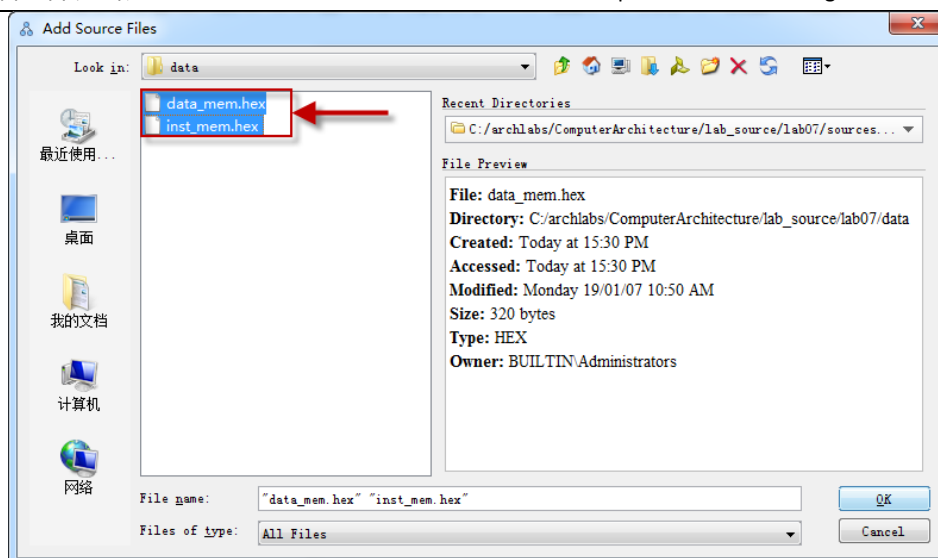


图 2-21 选择 hex 文件

选中 Scan and add RTL include files into project 和 Copy sources into project，点击 Finish 把文件加入到工程，如图 2-22 所示：

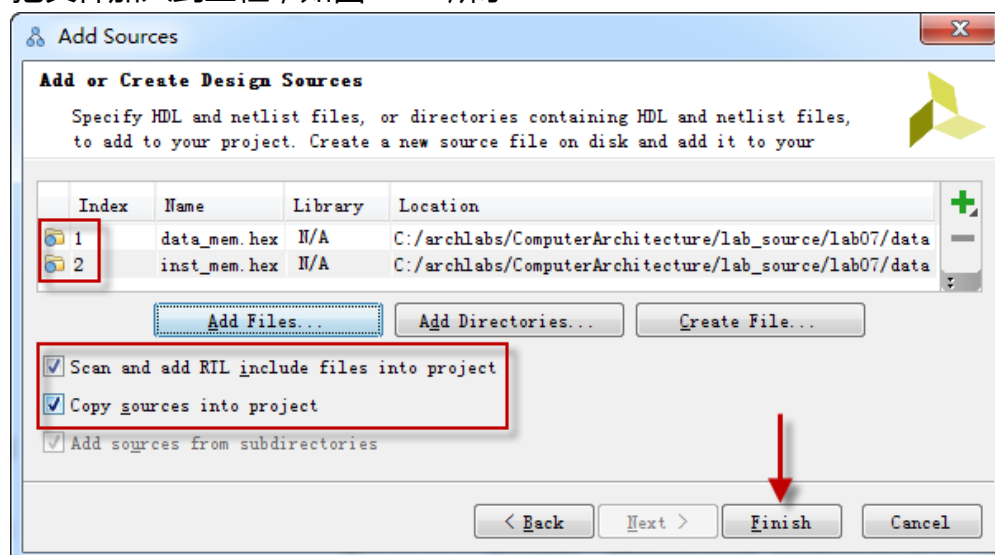



图 2-22 增加到工程

点击 “+” 展开 Unknown，在 data\_mem.hex 上点击右键，点击 Set File Type，如图 2-23 所示：

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	17 of 35
	作者 Joseph Xu	修改日期 2019/1/23	公开	

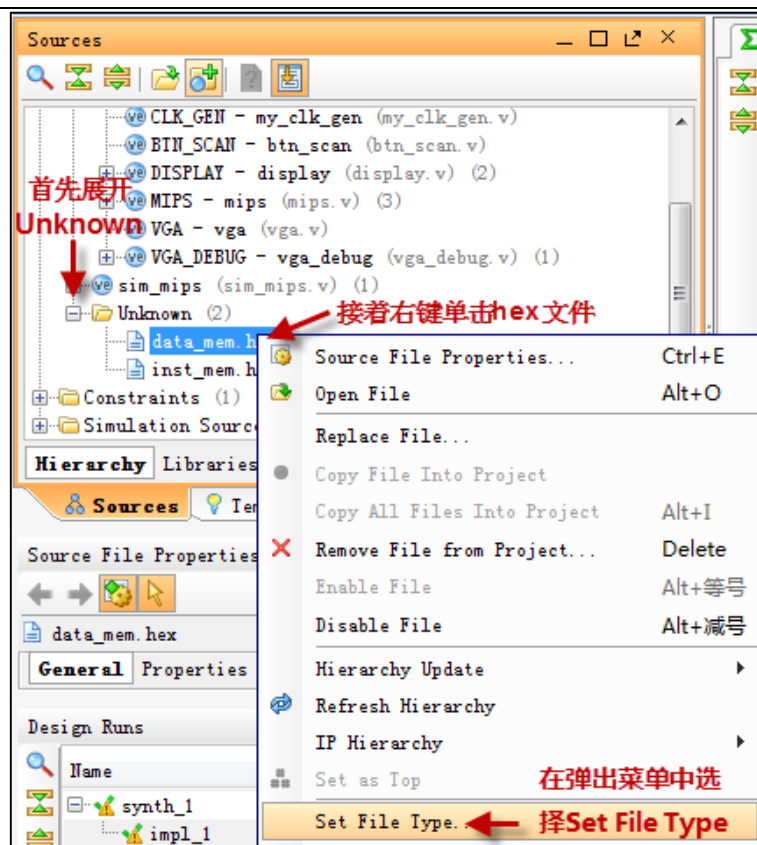


图 2-23 设置文件类型

选择 Data Files，点击 OK，如图 2-24 所示：

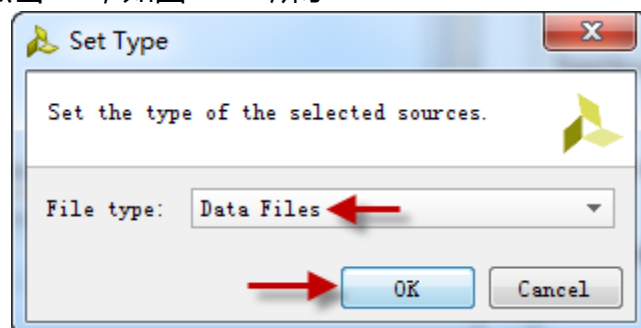



图 2-24 设置类型为 Data Files

如图 2-25 所示，同样设置 inst\_mem.hex 的类型。

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	18 of 35
	作者 Joseph Xu	修改日期 2019/1/23	公开	

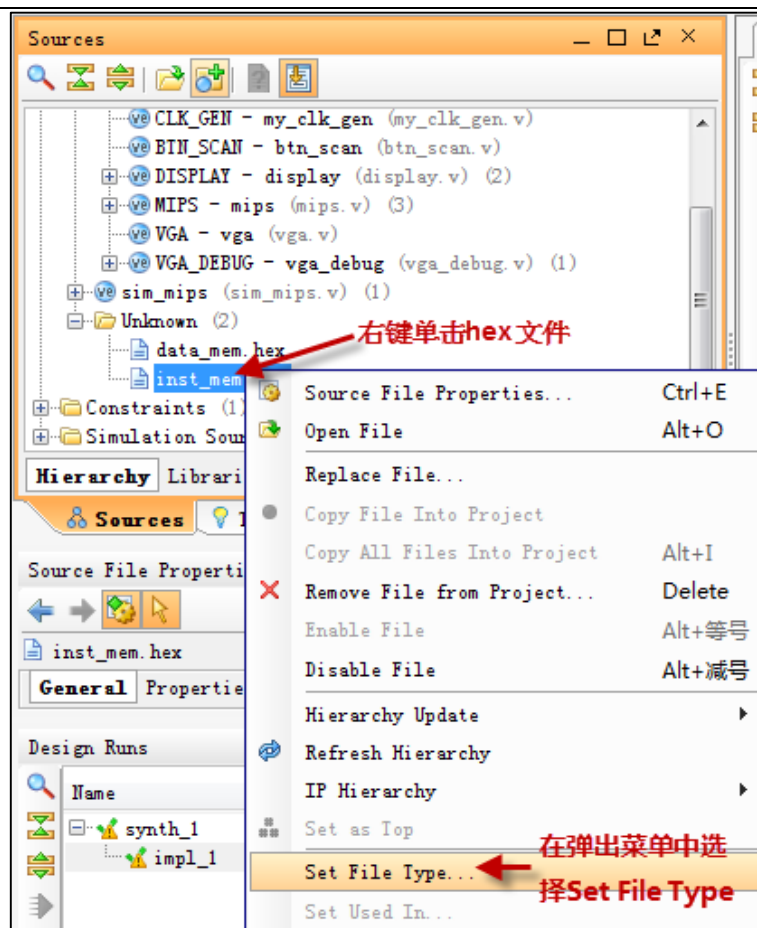


图 2-25 设置文件类型

设置为 Data Files，如图 2-26 所示：

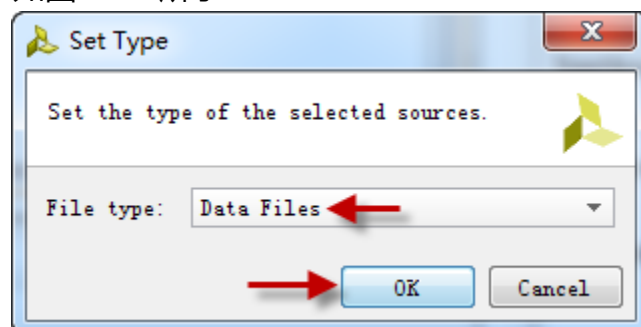



图 2-26 设置类型为 Data Files

在 data\_mem.hex 上点击右键，点击 Refresh Hierarchy 更新视图，如图 2-27 所示：

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	19 of 35
	作者	修改日期	公开	
	Joseph Xu	2019/1/23		

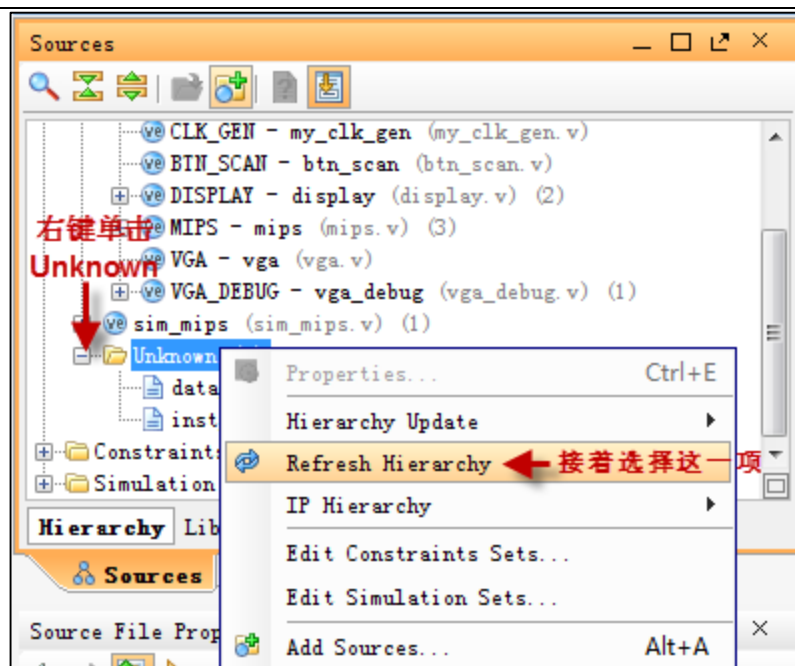


图 2-27 Refresh Hierarchy

如图 2-28 所示，现在 data\_mem.hex 和 inst\_mem.hex 都在 Data Files 下了，这两个文件包含指令存储器和数据存储器的初始化信息。

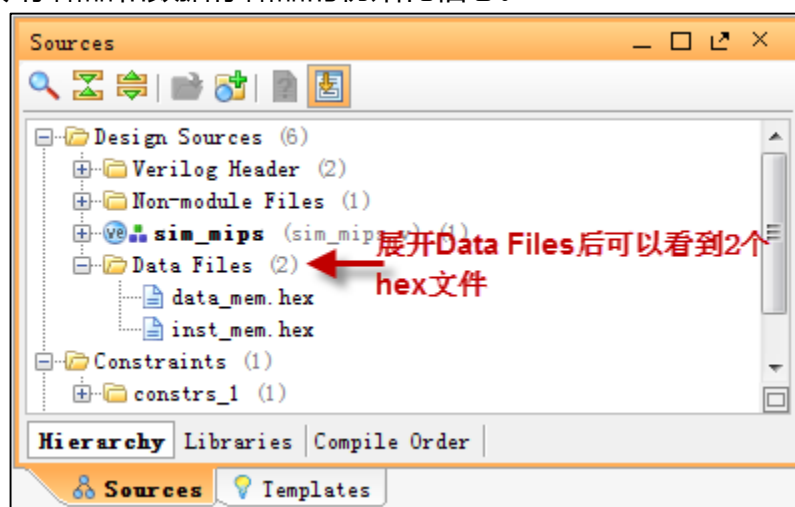



图 2-28 Data Files

添加的文件列表应如图 2-29 所示，**请仔细比对该图，确保所有新文件都已经添加！！**

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	20 of 35
	作者 Joseph Xu	修改日期 2019/1/23	公开	

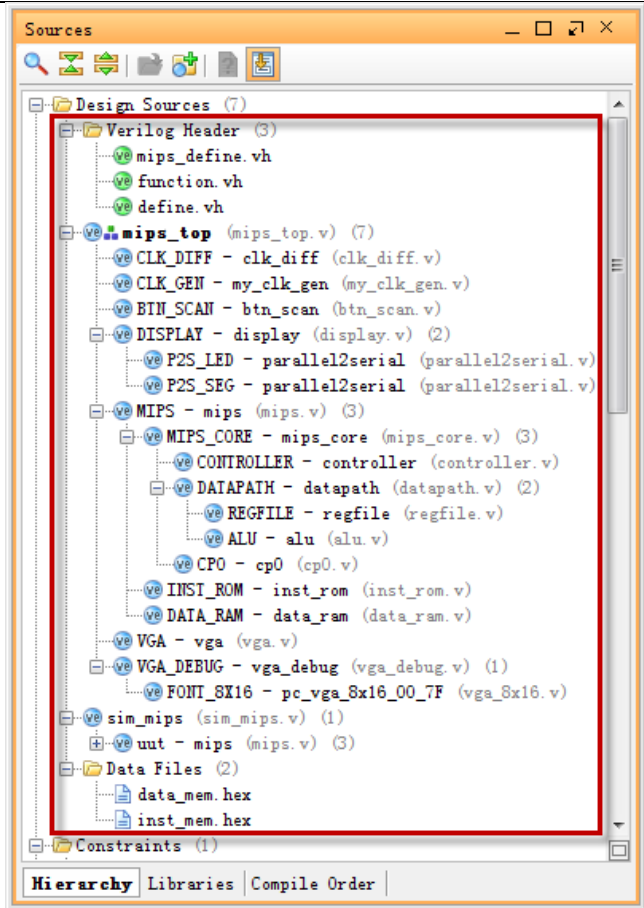



图 2-29 添加所有文件后的文件列表

2.2 功能仿真

如图 2-30 所示，在 Vivado 的 Sources 窗口的 Simulation Source 子项依次展开，然后将 sim\_mips 设置为仿真顶层：

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	21 of 35
	作者	修改日期	公开	
	Joseph Xu	2019/1/23		

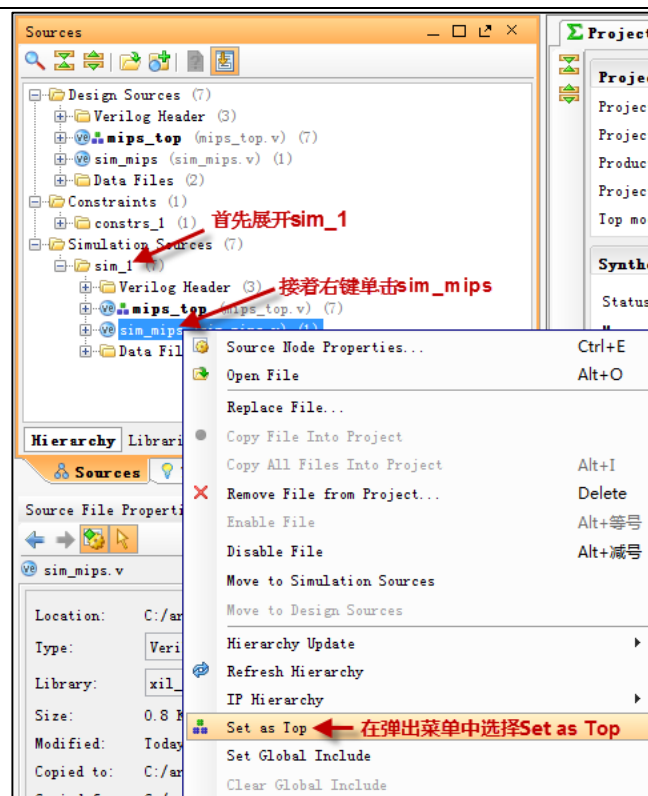


图 2-30 设置仿真顶层

点击 Run Simulation 下的 Run Behavioral Simulation，如图 2-31 所示：

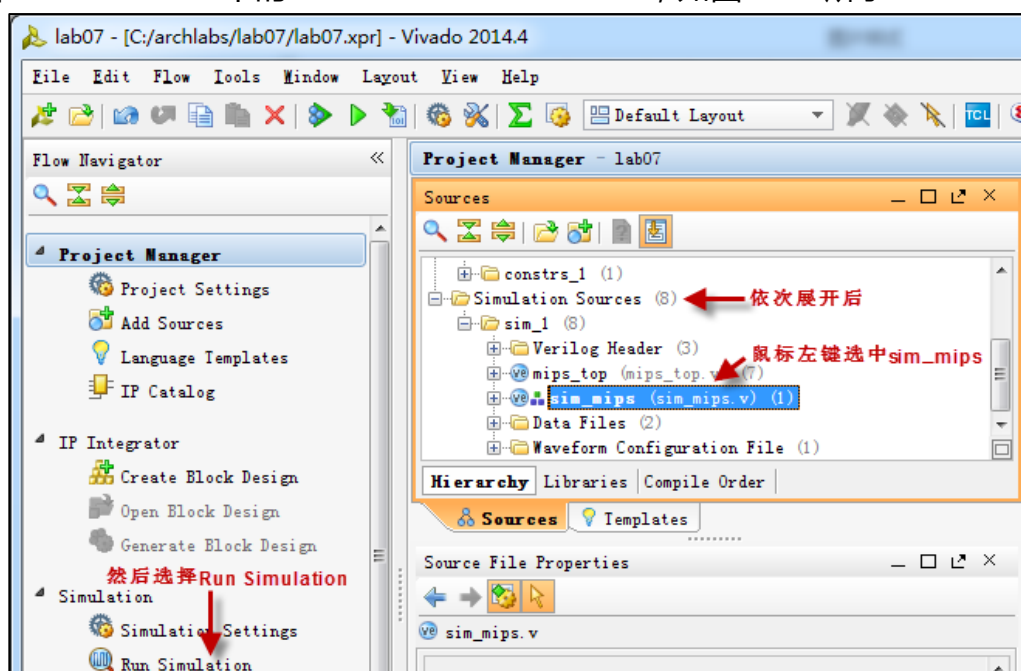



图 2-31 开始仿真

仿真编译成功后，进入仿真视图，如图 2-32 所示：

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	22 of 35
	作者	修改日期		
	Joseph Xu	2019/1/23	公开	

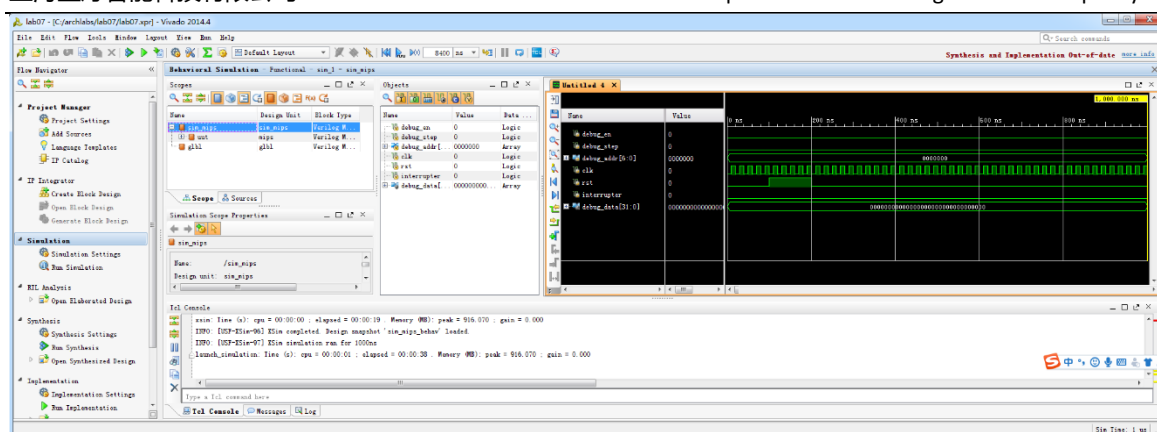



图 2-32 仿真视图

按照之前实验的仿真操作，将如下信号添加到波形列表中：

1. /sim\_mips/clk
2. /sim\_mips/rst
3. /sim\_mips/uut/MIPS\_CORE/DATAPATH/if\_rst
4. /sim\_mips/uut/MIPS\_CORE/DATAPATH/if\_en
5. /sim\_mips/uut/MIPS\_CORE/DATAPATH/if\_valid
6. /sim\_mips/uut/MIPS\_CORE/DATAPATH/inst\_addr[31:0]
7. /sim\_mips/uut/MIPS\_CORE/DATAPATH/inst\_data[31:0]
8. /sim\_mips/uut/MIPS\_CORE/DATAPATH/inst\_ren
9. /sim\_mips/uut/MIPS\_CORE/DATAPATH/inst\_stall
10. /sim\_mips/uut/MIPS\_CORE/DATAPATH/id\_rst
11. /sim\_mips/uut/MIPS\_CORE/DATAPATH/id\_en
12. /sim\_mips/uut/MIPS\_CORE/DATAPATH/id\_valid
13. /sim\_mips/uut/MIPS\_CORE/DATAPATH/inst\_addr\_id[31:0]
14. /sim\_mips/uut/MIPS\_CORE/DATAPATH/inst\_data\_id[31:0]
15. /sim\_mips/uut/MIPS\_CORE/DATAPATH/fwd\_a\_ctrl[1:0]
16. /sim\_mips/uut/MIPS\_CORE/DATAPATH/fwd\_b\_ctrl[1:0]
17. /sim\_mips/uut/MIPS\_CORE/CONTROLLER/reg\_stall
18. /sim\_mips/uut/MIPS\_CORE/DATAPATH/exe\_rst
19. /sim\_mips/uut/MIPS\_CORE/DATAPATH/exe\_en
20. /sim\_mips/uut/MIPS\_CORE/DATAPATH/exe\_valid
21. /sim\_mips/uut/MIPS\_CORE/DATAPATH/inst\_addr\_exe[31:0]
22. /sim\_mips/uut/MIPS\_CORE/DATAPATH/inst\_data\_exe[31:0]
23. /sim\_mips/uut/MIPS\_CORE/CONTROLLER/wb\_rst
24. /sim\_mips/uut/MIPS\_CORE/CONTROLLER/wb\_valid
25. /sim\_mips/uut/MIPS\_CORE/DATAPATH/wb\_wen\_wb
26. /sim\_mips/uut/MIPS\_CORE/DATAPATH/regw\_addr\_wb[4:0]
27. /sim\_mips/uut/MIPS\_CORE/DATAPATH/regw\_data\_wb[31:0]
28. /sim\_mips/uut/MIPS\_CORE/CP0/ir\_in
29. /sim\_mips/uut/MIPS\_CORE/CP0/oper[1:0]
30. /sim\_mips/uut/MIPS\_CORE/CP0/addr\_r[4:0]
31. /sim\_mips/uut/MIPS\_CORE/CP0/data\_r[31:0]

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	23 of 35
	作者 Joseph Xu	修改日期 2019/1/23	公开	



32. /sim\_mips/uut/MIPS\_CORE/CP0/addr\_w[4:0]
33. /sim\_mips/uut/MIPS\_CORE/CP0/data\_w[31:0]
34. /sim\_mips/uut/MIPS\_CORE/CP0/jump\_en
35. /sim\_mips/uut/MIPS\_CORE/CP0/jump\_addr[31:0]
36. /sim\_mips/uut/MIPS\_CORE/CP0/epcr[31:0]
37. /sim\_mips/uut/MIPS\_CORE/CP0/ehbr[31:0]
38. /sim\_mips/uut/MIPS\_CORE/DATAPATH/REGFILE/regfile[1:31][31:0]

添加后将所有 4 位以上宽度的信号都设置成 16 进制显示。添加并设置后的波形列表如图 2-33 所示：

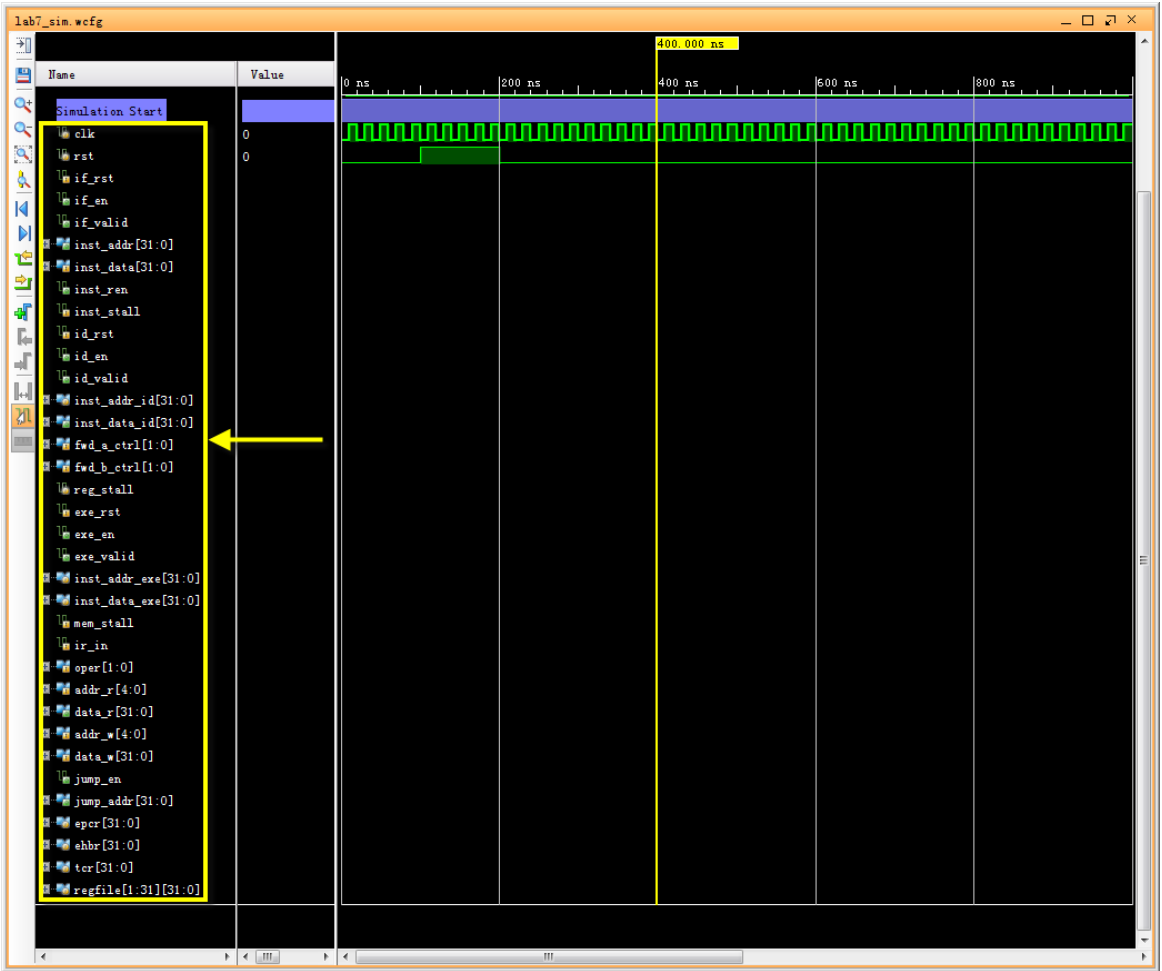



图 2-33 仿真所需要观察的所有信号

另外为了方便操作，可以直接加载预先提供的波形列表配置文件，具体操作如下：  
在仿真视图下，选择 File→Open Waveform Configuration，如图 2-34 所示：

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	24 of 35
	作者	修改日期	公开	
	Joseph Xu	2019/1/23		

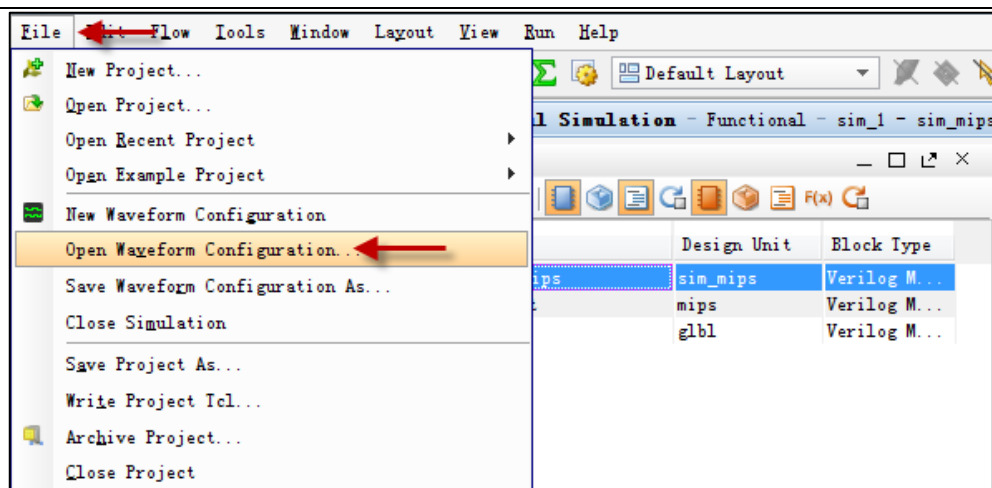


图 2-34 菜单中选择“Open Waveform Configuration”

切换到 C:\archlabs\ComputerArchitecture\lab\_source\lab07\simulation 目录，如图 2-35 所示：

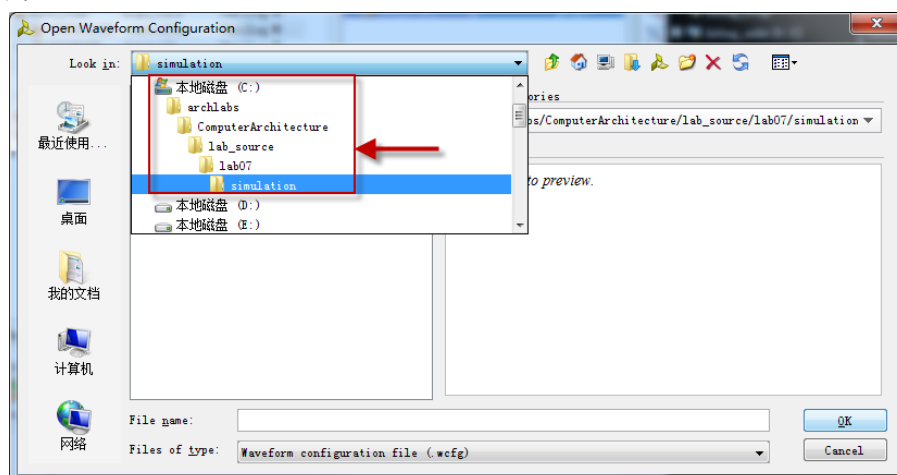


图 2-35 切换目录

选择 lab7\_sim.wcfg 文件，然后点击 OK。如图 2-36 所示：

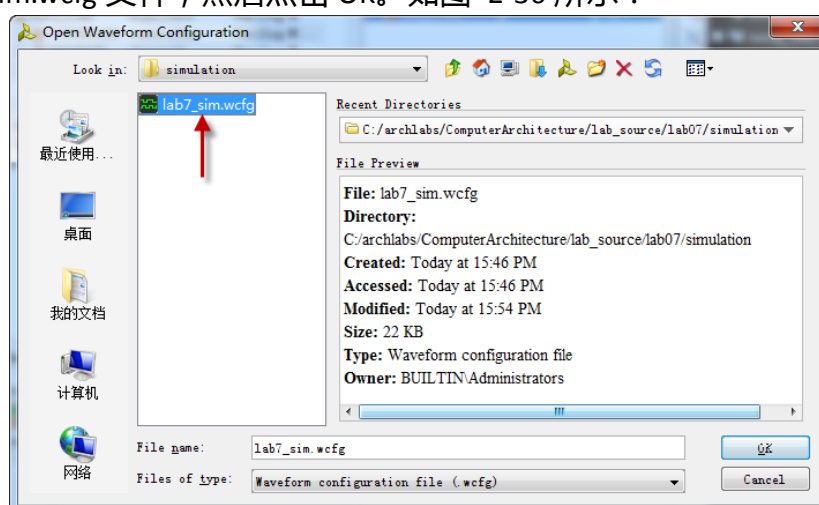



图 2-36 选择 wcfg 文件

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	25 of 35
	作者	修改日期	公开	
	Joseph Xu	2019/1/23		

添加后的信号如图 2-37 所示，：

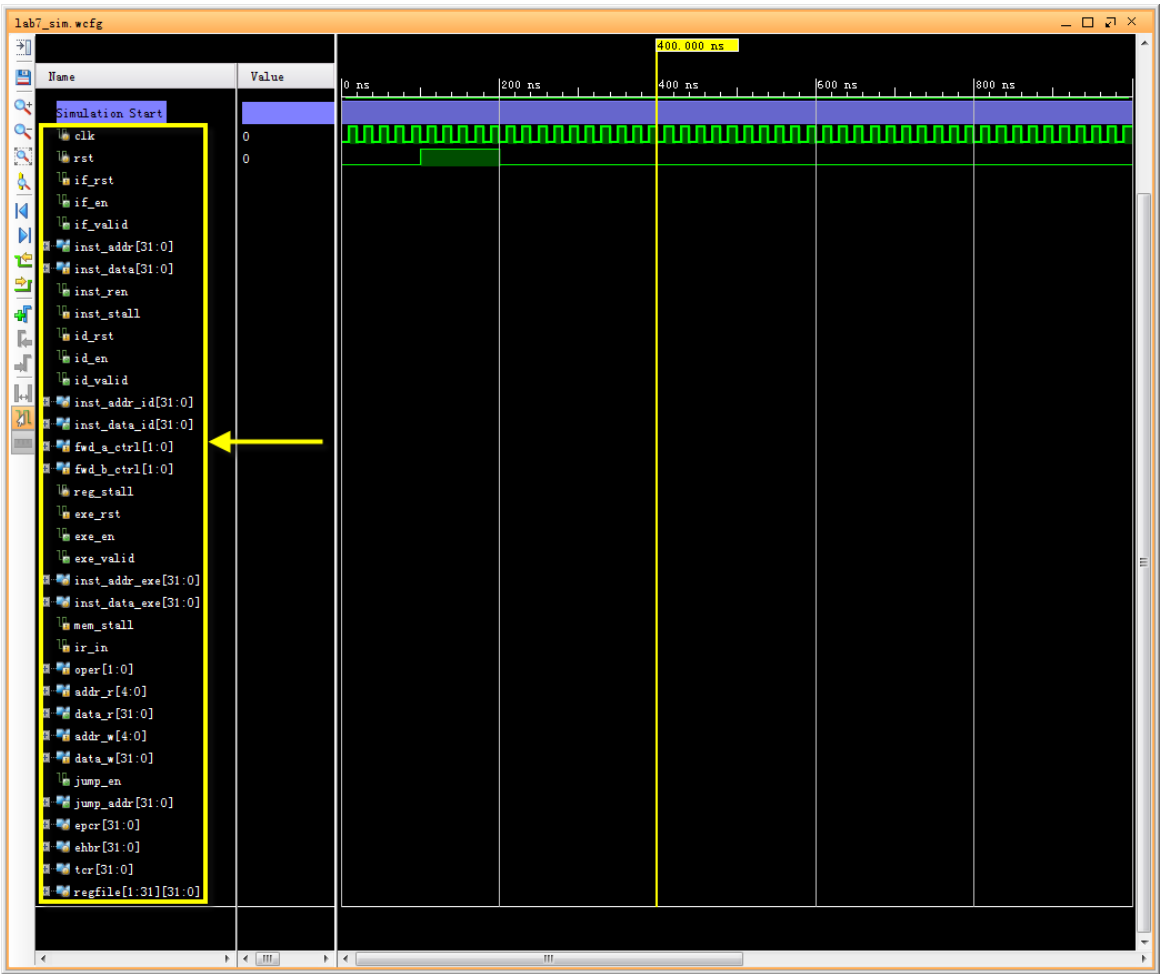


图 2-37 通过加载波形配置文件添加信号

增加信号后需要使用 Run 菜单下的 Restart，重新开始，如图 2-38 所示：

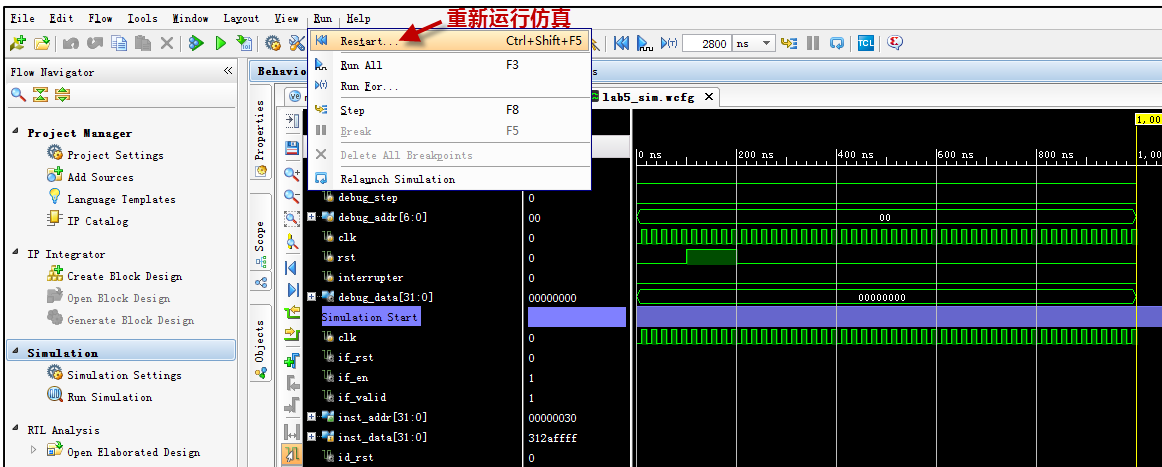



图 2-38 Restart Simulation

使用 Run for...仿真指定时间长的波形，如图 2-39 所示：

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	26 of 35
	作者	修改日期	公开	
	Joseph Xu	2019/1/23		

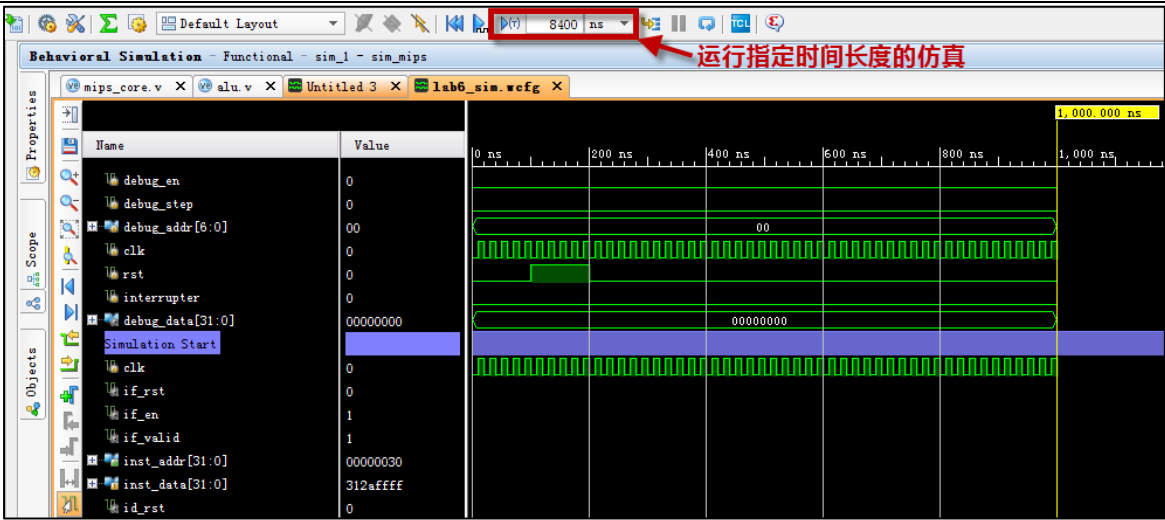



图 2-39 Run for...

通过键盘 Shift+ “-” 和 Shift+ “+” 可以对波形图进行缩放。  
将仿真出的波形和下面 2 张图对比，可以知道设计的电路是否正常工作。

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	27 of 35
	作者	修改日期	公开	
	Joseph Xu	2019/1/23		

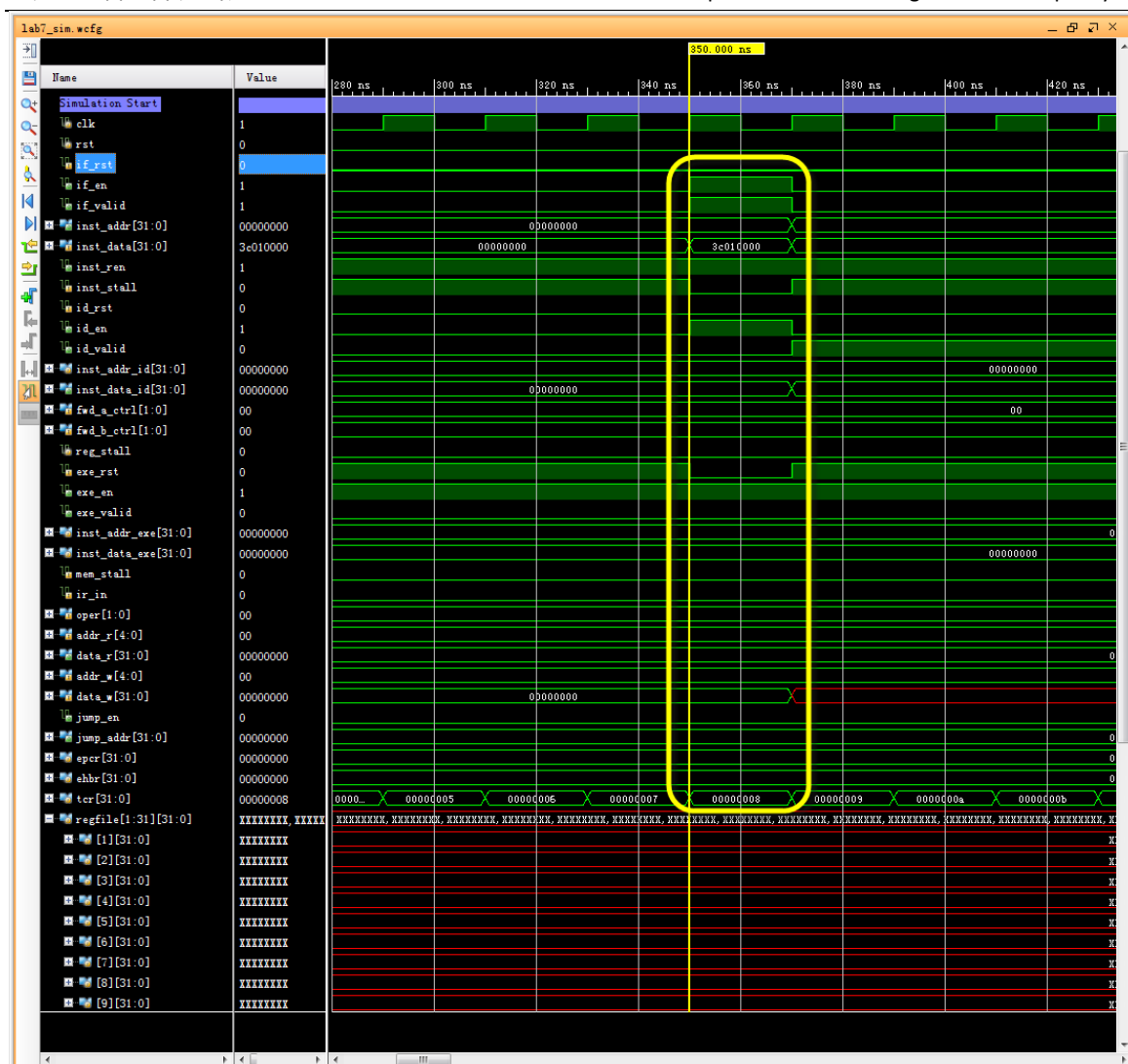



图 2-40 仿真结果 1

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	28 of 35
	作者	修改日期	公开	
	Joseph Xu	2019/1/23		

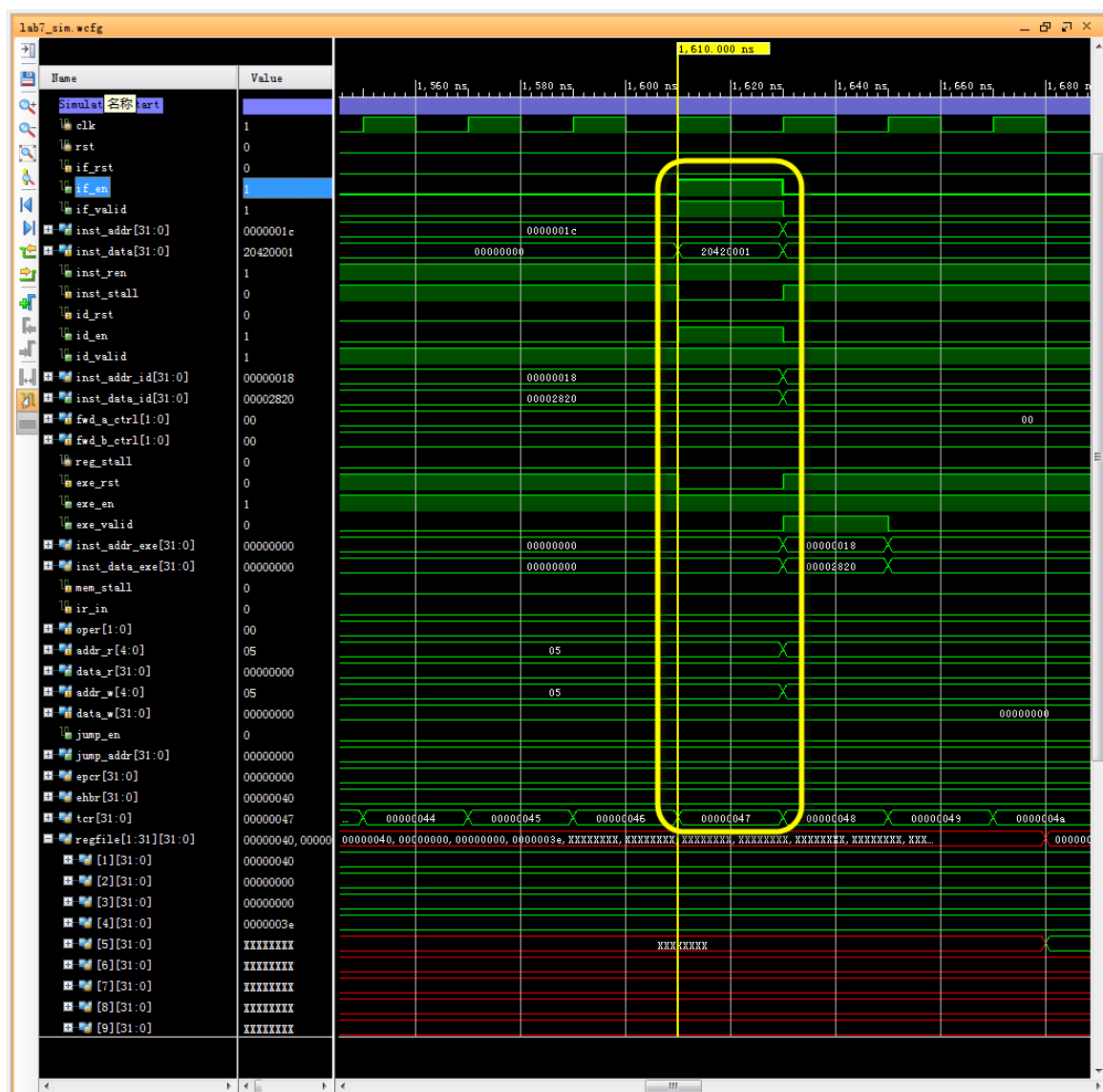


图 2-41 仿真结果 2

仿真结束后，在菜单中选择 File→Close Simulation，关闭仿真视图。如图 2-42 所示：

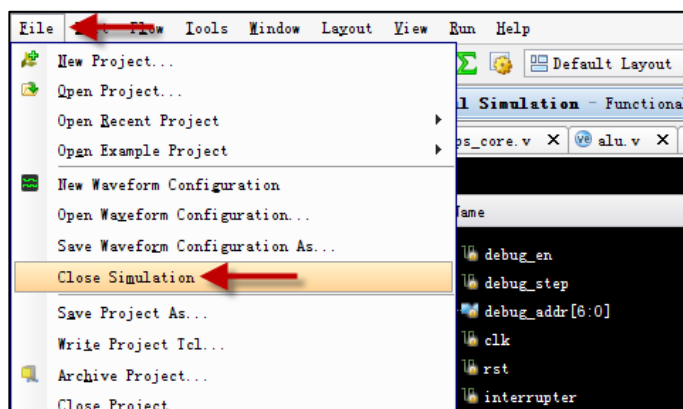



图 2-42 关闭仿真视图

	标题 Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	文档编号 XD-LAB-ARCH-007	版本   页 1.0 29 of 35
	作者 Joseph Xu	修改日期 2019/1/23	公开

## 2.3 生成 bitstream 文件

回到 Vivado 主界面后，确认 mips\_top.v 被设置为顶层文件后。点击 **Generate Bitstream** 生成 FPGA 配置文件。如图 2-43 所示：

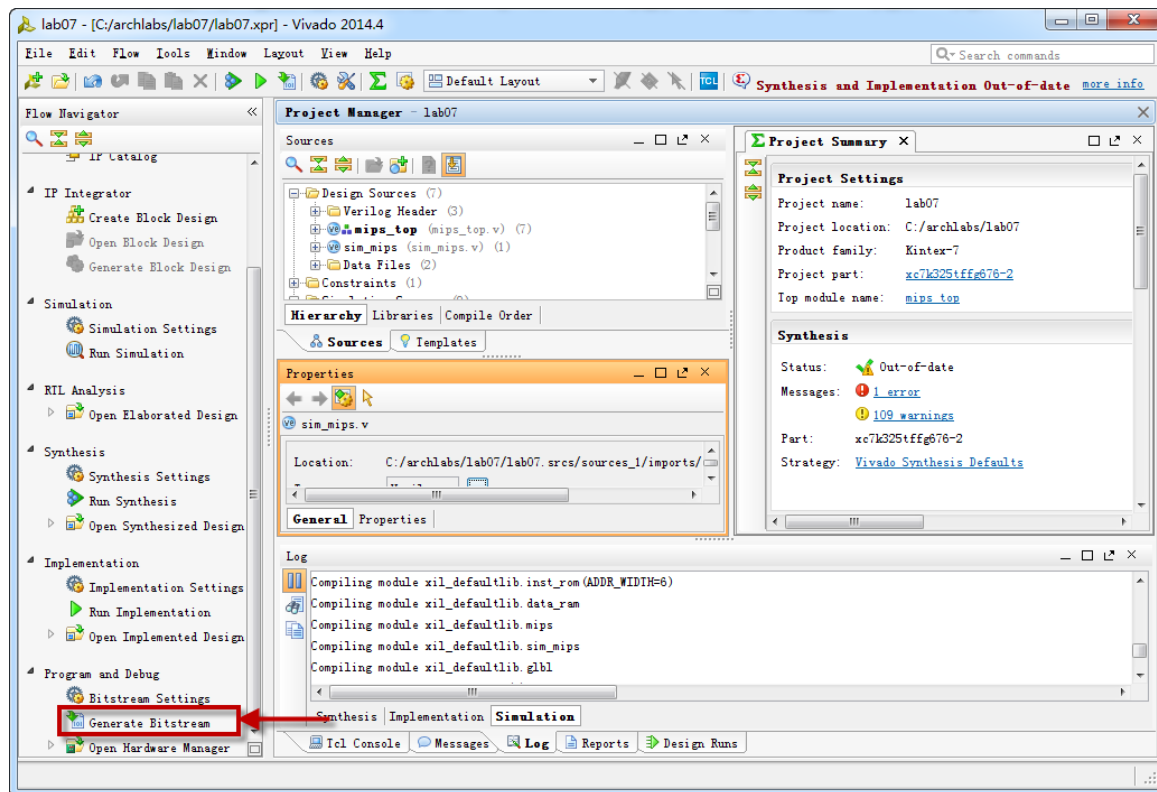


图 2-43 生成 bitstream

在询问对话框中选择 Yes，自动运行 synthesis 和 implementation，如图 2-44 所示：

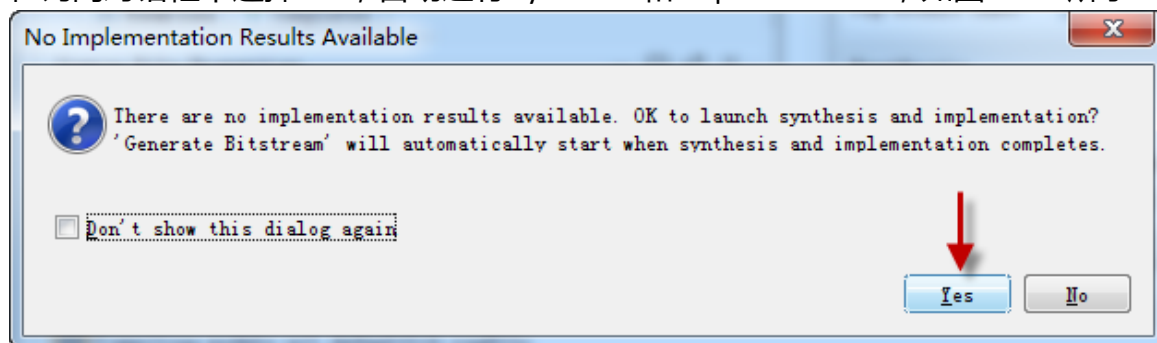



图 2-44 自动运行中间步骤

如图 2-45 所示，当显示 Bitstream Generation Completed 时表示文件成功生成，点击 **Cancel** 结束。

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	30 of 35
	作者 Joseph Xu	修改日期 2019/1/23	公开	



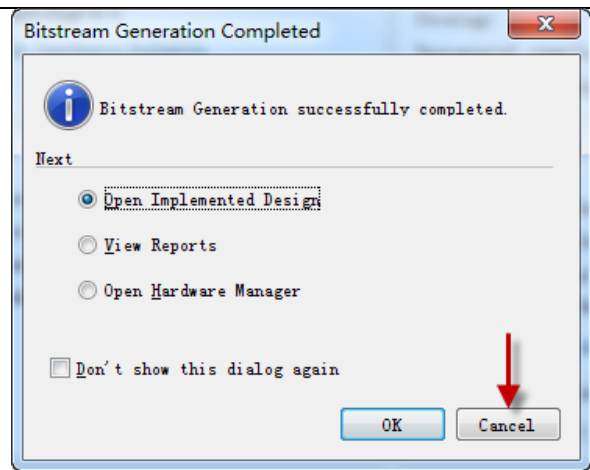


图 2-45 生成 bitstream 成功

2.4 下载验证

仿真完成后，下载 bit 文件到 SWORD4.0 开发平台查看真实效果。按图连接 SWORD4.0 的 12V 电源和 JTAG，然后打开开关。

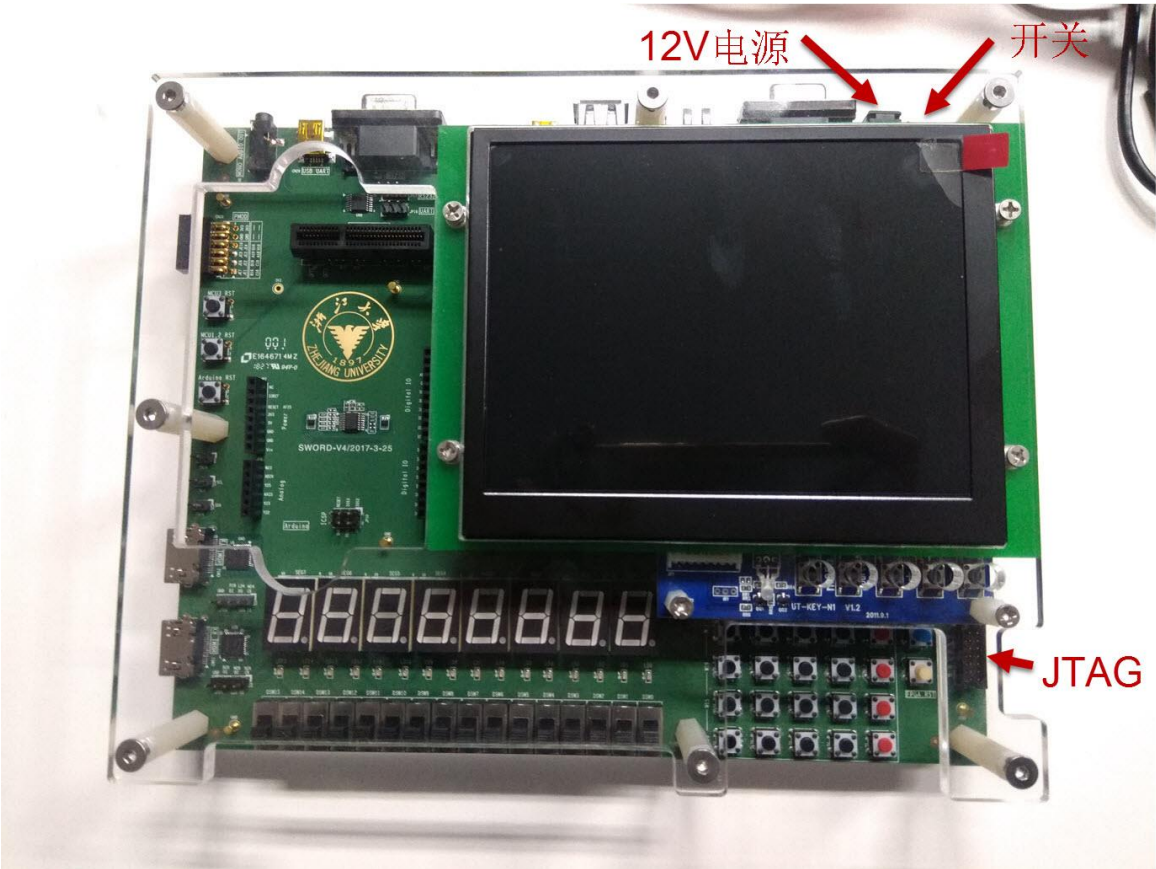



图 2-46 连接 SWORD4.0

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	31 of 35
	作者	修改日期	公开	
	Joseph Xu	2019/1/23		



SWORD4.0 的控制开关主要有拨码开关、键盘矩阵、复位按钮等。

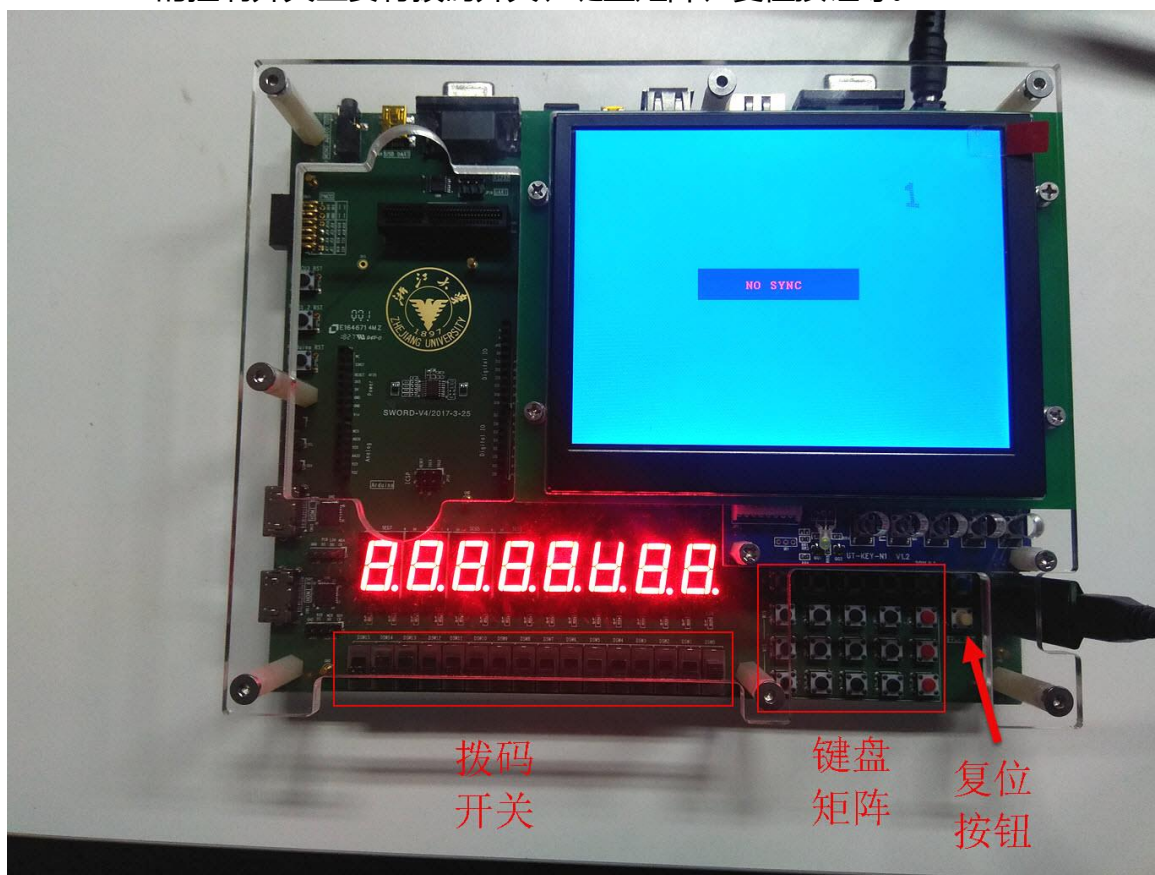


图 2-47 主要控制 IO

在 Vivado 中点击 **Open Hardware Manager**。

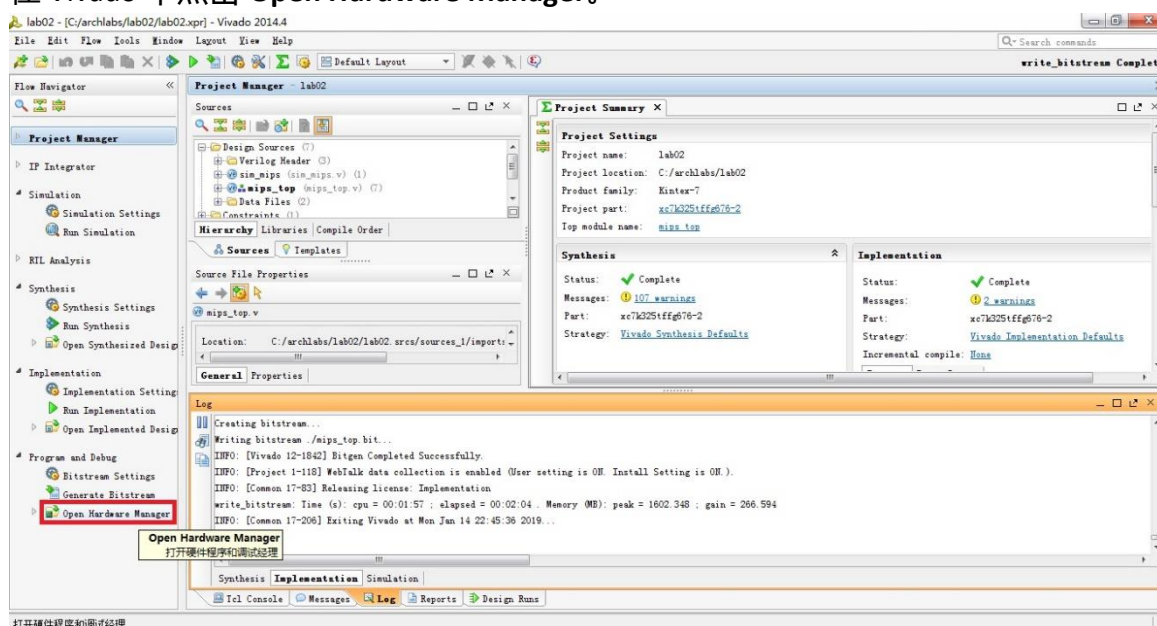



图 2-48 进入 Hardware Manager

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	32 of 35
	作者	修改日期	公开	
	Joseph Xu	2019/1/23		

点击 **Open target**，然后点击 **Auto Connect**。

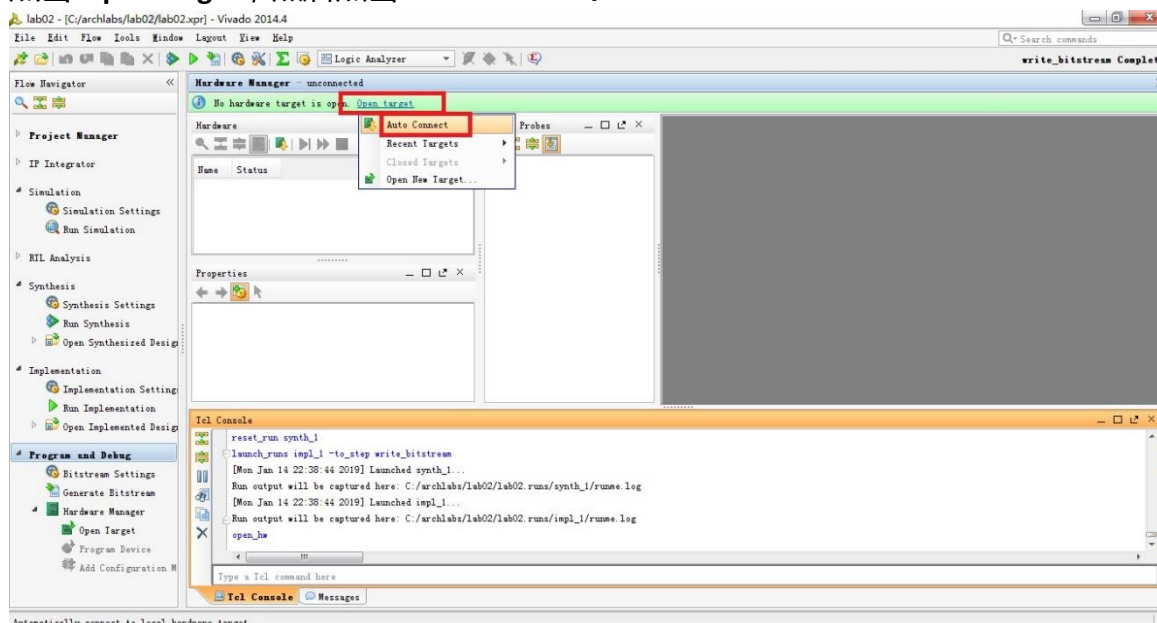


图 2-49 自动连接到 SWORD4.0

右键点击找到的 **xc7k325t\_0**，选择 **Program Device**。

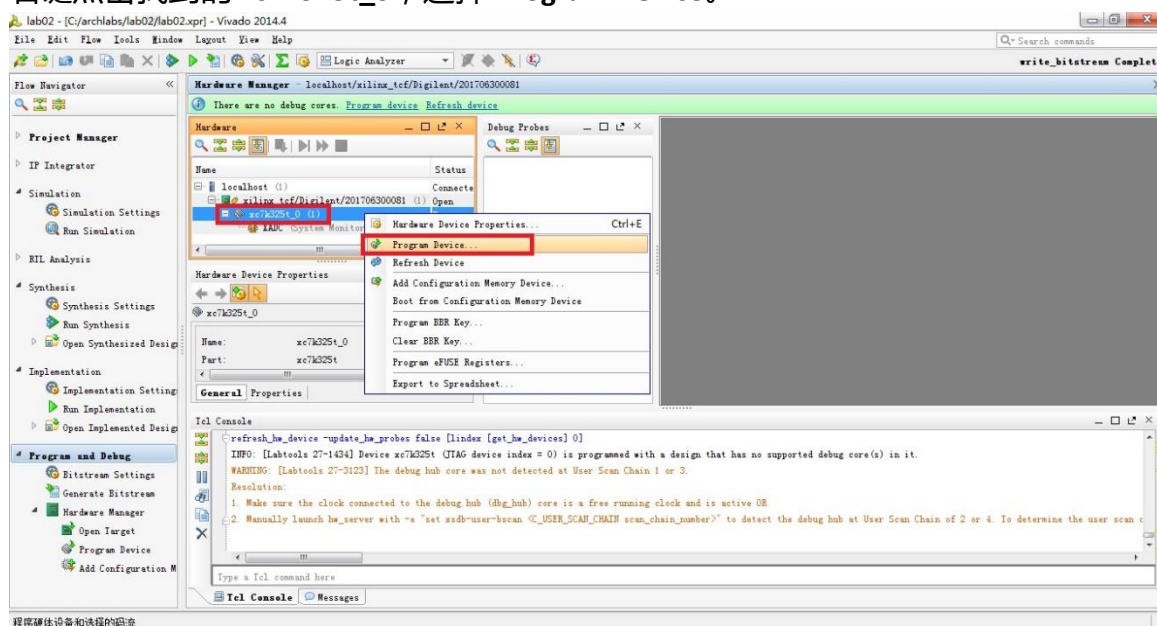



图 2-50 Program Device

点击 **Program** 下载 bit 文件。

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	33 of 35
	作者	修改日期		
	Joseph Xu	2019/1/23	公开	



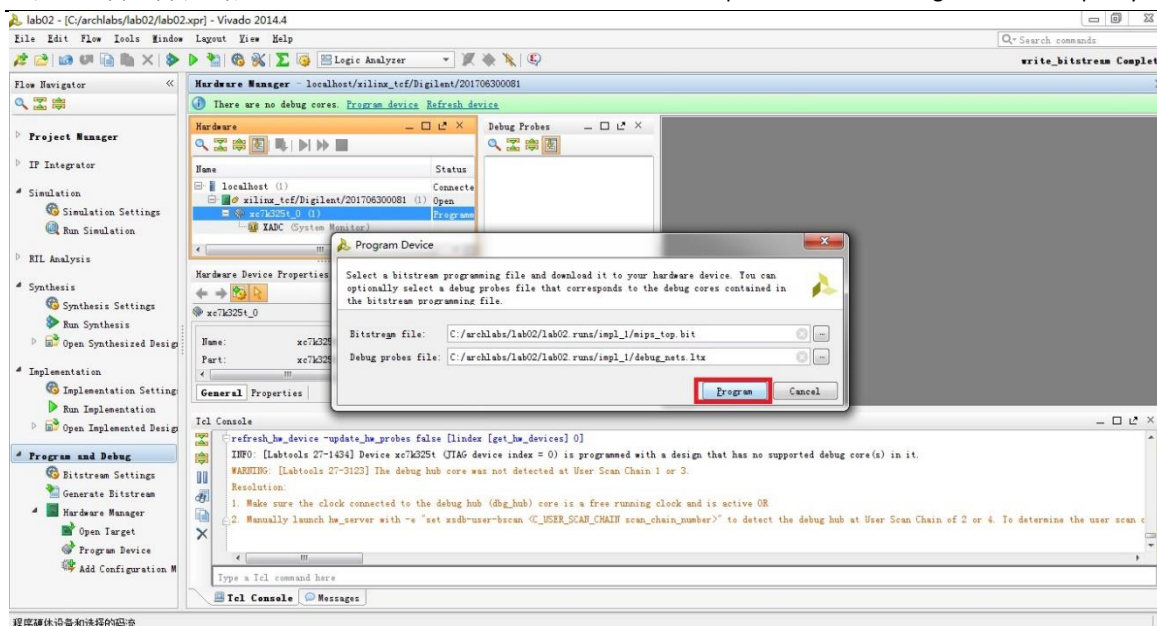


图 2-51 下载 bit 文件

下载完成后 SWORD4.0 上的反应，液晶屏上会显示调试信息。

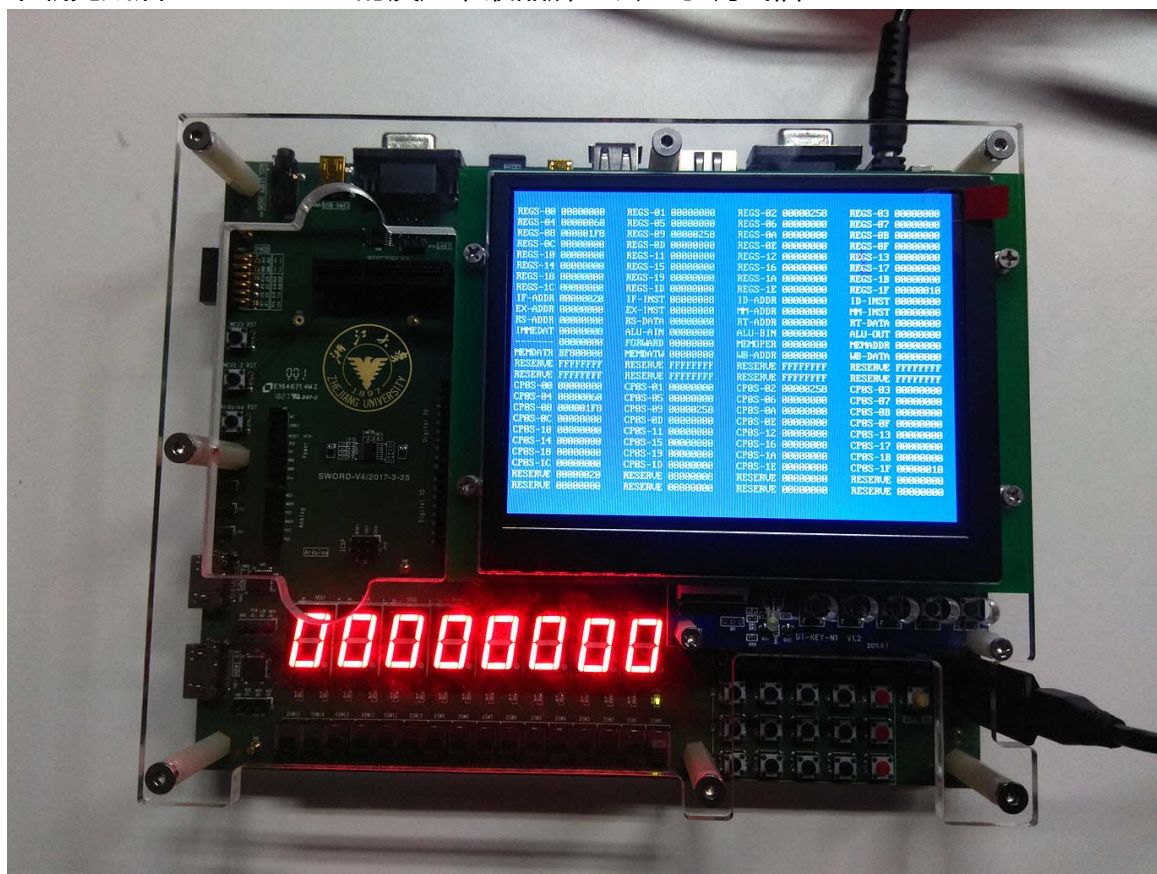




图 2-52 运行效果

通过将拨码开关 DSW0 拨向上，可以开启单步模式。之后每按一下键盘矩阵左下角的 BTNX4Y0，CPU 就执行一条指令。按复位键 FPGA RST 按钮回到初始状态。

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	34 of 35
	作者 Joseph Xu	修改日期 2019/1/23	公开	

在单步模式下，通过每一步液晶屏显示的数据与仿真波形的对比，查看流水线 cpu 是否正常工作。

	标题	文档编号	版本	页
	Lab7: Lab7: Pipelined CPU Accessing Mem in Multiple Cycle	XD-LAB-ARCH-007	1.0	35 of 35
	作者	修改日期	公开	
	Joseph Xu	2019/1/23		